Tarek Sobh
Editor

Innovations and Advanced Techniques in Computer and Information Sciences and Engineering

Springer

Innovations and Advanced Techniques in Computer and Information Sciences and Engineering

Innovations and Advanced Techniques in Computer and Information Sciences and Engineering

Edited by

Tarek Sobh
University of Bridgeport
CT, USA

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 978-1-4020-6267-4 (HB)
ISBN 978-1-4020-6268-1 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springer.com

Printed on acid-free paper

To Nihal, Omar, Haya, Sami and Adam

Table of Contents

Preface

This book includes Volume I of the proceedings of the 2006 International Conference on Systems, Computing Sciences and Software Engineering (SCSS). SCSS is part of the International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE 06). The proceedings are a set of rigorously reviewed world-class manuscripts presenting the state of international practice in Innovations and Advanced Techniques in Computer and Information Sciences and Engineering.

SCSS 06 was a high-caliber research conference that was conducted online. CISSE 06 received 690 paper submissions and the final program included 370 accepted papers from more than 70 countries, representing the six continents. Each paper received at least two reviews, and authors were required to address review comments prior to presentation and publication.

Conducting SCSS 06 online presented a number of unique advantages, as follows:

- All communications between the authors, reviewers, and conference organizing committee were done on line, which permitted a short six week period from the paper submission deadline to the beginning of the conference.
- PowerPoint presentations, final paper manuscripts were available to registrants for three weeks prior to the start of the conference
- The conference platform allowed live presentations by several presenters from different locations, with the audio and PowerPoint transmitted to attendees throughout the internet, even on dial up connections. Attendees were able to ask both audio and written questions in a chat room format, and presenters could mark up their slides as they deem fit
- The live audio presentations were also recorded and distributed to participants along with the power points presentations and paper manuscripts within the conference DVD.

The conference organizers and I are confident that you will find the papers included in this volume interesting and useful.

Tarek M. Sobh, Ph.D., PE
Bridgeport, Connecticut
June 2007

Acknowledgements

The 2006 International Conference on Systems, Computing Sciences and Software Engineering (SCSS) and the resulting proceedings could not have been organized without the assistance of a large number of individuals. SCSS is part of the International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE). I had the opportunity to co-found CISSE in 2005, with Professor Khaled Elleithy, and we set up mechanisms that put it into action. Andrew Rosca wrote the software that allowed conference management, and interaction between the authors and reviewers online. Mr. Tudor Rosca managed the online conference presentation system and was instrumental in ensuring that the event met the highest professional standards. I also want to acknowledge the roles played by Sarosh Patel and Ms. Susan Kristie, our technical and administrative support team.

The technical co-sponsorship provided by the Institute of Electrical and Electronics Engineers (IEEE) and the University of Bridgeport is gratefully appreciated. I would like to express my thanks to Prof. Toshio Fukuda, Chair of the International Advisory Committee and the members of the SCSS Technical Program Committee including: Abdelaziz AlMulhem, Alex A. Aravind, Ana M. Madureira, Mostafa Aref, Mohamed Dekhil, Julius Dichter, Hamid Mcheick, Hani Hagras, Marian P. Kazmierkowski, Low K.S., Michael Lemmon, Rafa Al-Qutaish, Rodney G. Roberts, Sanjiv Rai, Samir Shah, Shivakumar Sastry, Natalia Romalis, Mohammed Younis, Tommaso Mazza, and Srini Ramaswamy.

The excellent contributions of the authors made this world-class document possible. Each paper received two to four reviews. The reviewers worked tirelessly under a tight schedule and their important work is gratefully appreciated. In particular, I want to acknowledge the contributions of the following individuals: Khaled Elleithy, Yongsuk Cho, Michael Lemmon, Rafa Al-Qutaish, Yaser M. A. Khalifa, Mohamed Dekhil, Babar Nazir, Khaled Hayatleh, Mounir Bousbia-Salah, Rozlina Mohamed, A.Sima Etner-Uyar, Hussein Abbass, Ahmad Kamel, Emmanuel Udoh, Rodney G. Roberts, Vahid Salmani, Dongchul Park, Sergiu Dumitriu, Helmut Vieritz, Waleed Al-Assadi, Marc Wilke, Mohammed Younis, John Zhang, Feng-Long Huang, Natalia Romalis, Hamid Mcheick, Minkoo Kim, Khaled Rasheed, Chris Panagiotakopoulos, Alex Aravind, Dinko Gichev, Dirk Mueller, Andrew Vincent, Ana Madureira, Abhilash Geo Mathews, Yu Cai, Spyros Kazarlis, Liu Xia, Pavel Osipov, Hamad Alhammady, Fadel Sukkar, Jorge Loureiro, Hemant Joshi, Hossam Fahmy, Yoshiteru Ishida, Min Jiang, Vien Ngo Anh, Youming Li, X. Sheldon Wang, Nam Gyu Kim, Vasso Stylianou, Tommaso Mazza, Radu Calinescu, Nagm Mohamed, Muhammad Ali, Raymond Wu, Mansour Tahernezhadi, Trevor

Carlson, Sami Habib, Vikas Vaishnav, Vladimir Avdejenkov, Volodymyr Voytenko, Vygantas Petrauskas, Shivakumar Sastry, U. B. Desai, Julius Dichter, Hani Hagras, Giovanni Morana, Mohammad Karim, Thomas Nitsche, Rosida Coowar, Anna Derezinska, Amala Rajan, Aleksandras Vytautas Rutkauskas, A. Ismail, Mostafa Aref, Ahmed Abou-Alfotouh, Damu Radhakrishnan, Sameh ElSharkawy, George Dimitoglou, Marian P. Kazmierkowski, M . Basel Al-Mourad, Ausif Mahmood, Nawaf Kharma, Fernando Guarin, Kaitung Au, Joanna Kolodziej, Ugur Sezerman, Yujen Fan, Zheng Yi Wu, Samir Shah, Sudhir Veerannagari, Junyoung Kim and Sanjiv Rai.

Tarek M. Sobh, Ph.D., PE
Bridgeport, Connecticut
June 2007

Mobile Robot Localization using Soft-reduced Hypotheses Tracking

L. Banjanović-Mehmedović*, I. Petrović† and E. Ivanjko†
* University of Tuzla, Faculty of Electrical Engineering, Tuzla, Bosnia and Herzegovina
† University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb, Croatia

***Abstract*-Mobile robot localization is the problem of determining the pose (position and orientation) of a mobile robot under complex measurement uncertainties. The Soft-reduced Hypotheses Tracking algorithm introduced here is based on the modified multiple model and exploits a soft gating of the measurements to reduce the computational requirements of the approach. The position part is based on an *x*- and *y*-histograms scan matching procedure, where *x*- and *y*-histograms are extracted directly from local occupancy grid maps using probability scalar transformation. The orientation part is based on the proposed obstacle vector transformation combined with polar histograms. Proposed algorithms are tested using a Pioneer 2DX mobile robot.**

I. INTRODUCTION

The location awareness is important to many mobile robot applications. Localization techniques can be divided into local position tracking and global localization [1]. Local position tracking provides a new position estimate, given a previous position estimate and new information from proprioceptive and exteroceptive sensors. Kalman filter is the most common solution of the local localization problem. Global localization approach solves the uncertainty in the robot's pose, without initial pose information. It contains also the kidnapped and lost mobile robot problem.

A general framework to represent multiple position hypotheses and to reduce the mobile robot pose uncertainty is that of Markov localization [2]. Markov localization approach can solve those problems because multiple hypotheses are available. However, the accuracy of Markov localization is relatively low [1]. The more complex Multiple-Hypotheses Tracking (MHT) Scheme observes a multitude of different pose hypotheses, but it is difficult for implementation, because a large number of hypotheses may have to be maintained, which requires extensive computational resources. This leads to problems in a real-time implementation. Because of these difficulties, some other algorithms having smaller computational requirements were developed. One of this is Sequential Monte Carlo (SMC*)* or Condensation algorithm. Namely, the methods discussed above are mostly applicable to linear Gaussian state and observation models. As an alternative method for non-linear and/or non-Gaussian models is SMC, which has become a practical numerical technique to approximate the Bayesian tracking recursion. Monte Carlo localization exploits a sample-based method and computation burden of this method is low.

The here proposed **Soft-reduced Hypotheses Tracking (SRHT)** method combines the particle filtering technique with the philosophy behind the probabilistic data association filter PDAF [3]. In order to minimize the computational burden of the particle filter algorithm the number of particles is reduced. This is done by rejection of particles with sufficiently small likelihood values since they are not likely to be re-sampled using a soft-gating (SG) method. The basic idea of SG is to generate new particles that depend on the old state (cluttered measurements) and new measurements, starting with a set of samples approximately distributed according to the best hypothesis from initialization phase. The update step is repeated until a feasible likelihood value is received.

The state estimation problem refers to the selection of a good filter that copes with most of the situations in the application where it would be used. Among the estimation algorithms, the multiple model estimator (MME) is the best-known single-scan positional algorithm and is most widely used for the purpose of tracking maneuvering targets [4]. MME approach computes the state estimate that accounts for each possible current model using a suitable mixing of the previous model-conditioned estimates depending on the current model [5]. Amongst the available multiple models sophisticated techniques the multiple model estimator technique is the best cost-effective implementation and the modified form of this has here been chosen for mobile robot localization application [4].

The localization algorithm in our approach (SRHT) uses a hybrid representation of the environment, i.e. topological map with metric information. The node distances in the topological environment model is 1 (m). Using a combination of SG and modified MME estimator in scope of the implemented localization process, the computational cost is made independent on the size of the environment.

An electronic compass is often used for mobile robot orientation measurement, but it is sensitive to magnetic noise that comes from ferromagnetic objects or structures in the mobile robot environment, from the mobile robot body and the noise produced by its drive system. So it is good to avoid its usage and to develop an algorithm that estimates robot orientation using only sonar measurements. Then histogram-matching procedure given in [6] can be extended to estimate not only position, but also orientation of the mobile robot. While histograms for position tracking (*x*- and *y*-histograms) are extracted from local occupancy grid maps via *Probability*

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 1–6.

Scalar Transform (PST), polar histograms are obtained via *Obstacle Vector Transform* (OVT) [7]. The result of polar histograms comparison is used for mobile robot orientation correction and is crucial for reliable mobile robot localization when an electronic compass can't be used.

II. LOCALIZATION ALGORITHM STRUCTURE

Block scheme of the proposed localization algorithm is given in Fig.1.

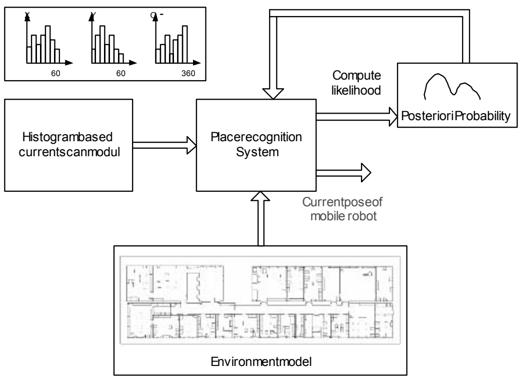


Fig. 1. Block scheme of the proposed localization algorithm.

To build the hybrid map, local occupancy grid maps containing environment metric information are stored at regular intervals of 1 (m). As the robot moves through its environment, sensor readings are obtained and transformed to new form in the *Histogram based current scan module*. Obtained *x*-, *y*- and *angle*-histograms with the pose hypothesis data are then passed to the *Place Recognition System* and matched with the activated hypotheses from the hybrid environment map.

The matching process is performed between an environment hypothesis and predicted hypothesis using the updated previous mobile robot pose. Only few hypotheses with maximum a posteriori probability are activated and updated, giving *predicted value* for the next step. The pose coordinates are updated according to the mobile robot movements measured by the wheel encoders of the mobile robot since the last pose update.

The hypothesis with maximum a posteriori probability within the set of activated hypotheses is considered as the *mobile robot current pose*. In this way, we obtain a reasonably accurate method of tracking the mobile robot pose and global localization of the mobile robot. The number of tracks can become too large in a dense environment. Although the number of associated hypotheses increases exponentially with an increase in the number of updated measurements, it is assumed in this approach that the *number of tracked hypotheses* is $N_T(k) \le N_V$, where N_V is the number of hypotheses that have to be tracked to achieve an acceptable pose tracking accuracy. From our experimental research we found out that 7 hypotheses fulfill the requirements for safe mobile robot navigation.

Fig. 2 presents large environment E and few clutters C_i, $i = 1,\dots N_T(k)$, in which mobile robot can be. Whenever the global position of the robot is uniquely determined, the huge state space of the estimation problem can be reduced to a small cube-clutter P centered around the robot's estimated pose.

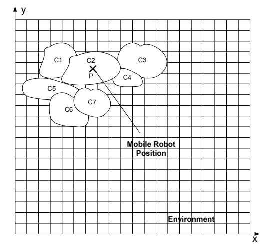


Fig. 2. Clutter centered around the robot's estimated position.

III. HISTOGRAM BASED CURRENT SCAN MODULE

A. X- and Y-histograms

In an occupancy grid map the mobile robot environment is presented with a grid in which each cell holds a certainty value that a particular area of space is occupied or empty [8]. The certainty value is based only on sonar sensor range readings. Each occupancy grid cell in our approach represents an area of 10 x 10 (cm^2) and is considered as being in one of three possible states: occupied (*O:* $P(c_{xy}) > 0.5$), empty (*E:* $P(c_{xy}) < 0.5$) and unknown (*U:* $P(c_{xy}) = 0.5$), depending on the corresponding probability of occupancy for that cell.

Each local grid map, consisting of 60 x 60 cells, is represented by three one-dimensional histograms. Namely, on top of the constructed local occupancy grid we can get three types of histograms: *x*-, *y*- and *angle*-histogram. Both *x*- and *y*-histograms are consisted of three one-dimensional arrays, which are obtained by adding up the total number of occupied, empty and unknown cells in each of the 60 rows or columns respectively (*Probability Scalar Transform*).

Fig. 3.a) presents part of mobile robot environment before applying the Probability Scalar Transformation. Fig. 3.b) presents x and y histogram of current mobile robot scan.

Local Occupancy Grid

0	1	0	0,5	1	0	0,5	0	0,5	0
0	0,5	0,5	0	1	0	1	1	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0,5	0	0	1	0	0
0	1	0	0	1	1	0	0,5	0	0
0	0,5	0	0	0,5	1	0	0	0	0
1	0,5	0	0	0	0	0	0	0	0
0	1	0	0,5	1	0,5	0	0	0,5	0
0	0	0	0	0	0	0,5	1	0	0

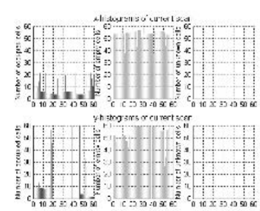


Fig. 3. a) Occupancy grid map; b) *x*- and *y*- histograms obtained from Probability Scalar Transform of current mobile robot scan.

B. Polar histograms

Polar histograms are used in vision systems [9], and as vector histograms for obstacle avoidance [10]. In our localization approach we extend their use to mobile robot orientation estimation using *Obstacle Vector Transform* (OVT). The idea is to first detect obstacles in the nearby mobile robot environment, present them using obstacle vectors and then to construct the polar histogram using only local sonar range readings (Fig. 4.a)).

Namely, a one-dimensional *polar histogram* is constructed around the robot's momentary location, using the OVT, which maps the local occupancy grid onto polar histogram. The histogram comprises n angular sectors of width α. Thus, for $\alpha = 1$ (º) there are $n = 360$ sectors. Each sector, holding a value of nearest obstacle distance, represents the polar obstacle vector in the direction that corresponds to it. The angles between the vector obstacle segments and positive x-axis weighted with obstacle vector length form then the polar histogram (Fig. 4.b)).

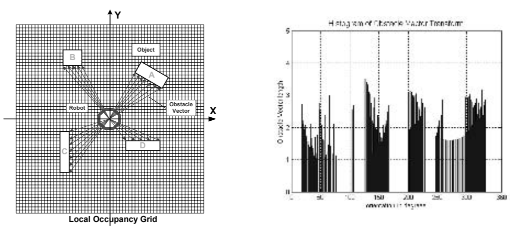


Fig. 4. a) Obstacle Vector Transform of current mobile robot scan; b) Polar histogram obtained with OVT.

The scans converting to histograms before matching is performed in order to reduce computation complexity. Comparison of polar or angle histograms in this way are possible solutions for robot orientation correction in environments with significant magnetic interference and a worth alternative to the use of magnetic compass.

C. Histogram matching

Matching scores of stored histogram (nodes of hybrid map) and recognition-translated histogram of current place are calculated for x- and y-histograms as [6]:

$$M\left(H_i, H_k\right) = SCALE * \sum_j \left[\min\left(O_j^i, O_j^k\right) + \min\left(E_j^i, E_j^k\right) + \min\left(U_j^i, U_j^k\right)\right]. \quad (1)$$

where Oj, Ej, Uj refer to the number of occupied, empty and unknown cells, contained in the j-th element of histogram H and SCALE scaling parameter.

For each of these hypotheses, the likelihood of sensor model $L(S|h_i)$ is calculated as the strength of the match between the current and stored histograms for each place hypothesis h_i:

$$L\left(S \mid h_i\right) \propto M_x^{i*} \times M_y^{i*}. \quad (2)$$

where are M_x-matching score of x-histogram, M_y-matching score of y-histogram, M_x^{i*} and M_y^{i*} are the best match scores, produced by the best matching alignment between histogram of chosen hypothesis h_i and translated histogram for the current place.

Whenever the robot moves (new measurements are obtained every 0,5 m), we can obtain the coordinates of each $H_i(k)$, i = 1,… N_C, (where $1 \le N_C \le N$) as:

$$x_{Mi}\left(k\right) = x_i + \Delta s_{xi}, \quad (3)$$

$$y_{Mi}\left(k\right) = y_i + \Delta s_{yi}, \quad (4)$$

which are calculated from coordinates of the place (x_i, y_i) and the offset values Δs_{xi} and Δs_{yi} obtained from the histogram matching process.

Localization begins with an *initialization step* taking a sonar scan and matching with a set of location hypotheses. For each of these hypotheses, the likelihood of sensor model $L(S|H_i)$ is obtained using (2), and the coordinates $x_{Mi}\left(k-1\right)$, $y_{Mi}\left(k-1\right)$ are obtained using (3) and (4). The initial probability distribution is then calculated using the following equation:

$$\widehat{p}(k-1 \mid H_i) = \frac{L(S \mid H_i)}{\sum_{q=1}^{N} L(S \mid H_q)}. \quad (5)$$

IV. Place Recognition System

The proposal for all association hypotheses depends on the information available at the current time step $S(k)$ and the previous model-conditioned estimates $\mathcal{H}(k)$ as is depicted in Fig. 5. It presents in which way current position and variance can be calculated from previous estimate, expected result from robot motion and current input.

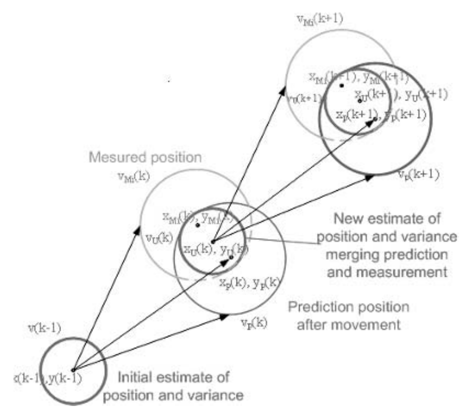


Fig. 5. Presentation estimate of position and variance

The proposed localization algorithm consists of three phases:

<u>*Predict phase*</u> – mobile robot pose is predicted according to the updated values in the previous step (k-1), measured displacement in x-axis and y-axis direction, and measured orientation change:

$$\overline{x}_i(k) = \hat{x}_i\left(k-1\right) + \Delta x, \quad (6)$$

$$\bar{y}_i(k) = \hat{y}_i(k-1) + \Delta y . \qquad (7)$$

Mobile robot orientation is predicted using updated value of orientation from previous step and orientation changes due to robot navigation:

$$\bar{\theta}(k) = \hat{\theta}(k-1) + \Delta\theta , \qquad (8)$$

where Δx, Δy and Δθ refer to the robot's own displacement in Cartesian space.

<u>*Data association or matching phase*</u> - a matching process is performed between cluttered environment hypotheses H_j, $j = 1,\dots,N_C$ and predicted hypotheses of updated previous mobile robot pose H_i , $i = 1, \dots,N_T(k)$.

$$L(H_j \mid H_i) \propto e^{-\eta\times\left\|\left(x_{Mj}(k),y_{Mj}(k)\right)-\left(\bar{x}_i(k),\bar{y}_i(k)\right)\right\|} \times \bar{p}(k-1 \mid H_i) , \qquad (9)$$

where the Gaussian function is used to model the noise in the mobile robot pose estimates and prior probability $\bar{p}(k-1 \mid H_i)$ is used to take the influence of particular prior hypothesis into account.

A common method of estimating nonrandom parameters is the maximum likelihood method that maximizes the likelihood function. This yields the maximum likelihood estimator (MLE):

$$\hat{l}^{ML}(L) = \arg\max_{l}\left[L(H_j \mid H_i)\right] . \qquad (10)$$

<u>*Data fusion or update phase*</u> – mobile robot pose is updated using values obtained in the matching phase according to the following equations:

$$\hat{x}_i(k) = \bar{x}_i^*(k) + K_1(k)\times\left(x_{Mi}(k) - \bar{x}_i^*(k)\right), \qquad (11)$$

$$\hat{y}_i(k) = \bar{y}_i^*(k) + K_1(k)\times\left(y_{Mi}(k) - \bar{y}_i^*(k)\right), \qquad (12)$$

$$K_1(k) = \frac{\bar{\sigma}_{i*}^2(k)}{\bar{\sigma}_{i*}^2(k) + \sigma_{Mi}^2(k)} , \qquad (13)$$

where $\bar{x}_i^*(k)$ and $\bar{y}_i^*(k)$ are the predicted values of N_T best hypotheses for which maximum likelihood estimator (MLE) estimators $\hat{l}^{ML}(L)$ have maximum values.

$\sigma_{Mi}(k)$ is scan matching variance for each hypothesis $H_i(k)$ and $\bar{\sigma}_i(k)$ is predicted values of variance from updated previous value $\bar{\sigma}_i(k-1)$, $i = 1,\dots N_T(k)$.

In our approach, the calculation χ^2_{TH} is used, because it gives good results in mobile robot orientation tracking:

$$M_\theta = \chi^2_{TH}(H(k), H(k-1)) = \sum_j \frac{(H_j(k) - H_j(k-1))^2}{H_j(k) + H_j(k-1)} , \qquad (14)$$

where $H(k)$ and $H(k$-1) are current and previous histograms, respectively.

All hypothetic robot orientations θ_j^{MLE} (orientation hypotheses) with equal minimal matching score,

$$\theta_j^{MLE}(M) = \arg\min_{M}\left[M_\theta\right] , \qquad (15)$$

obtained by angle histograms convolutions are used to determine the best orientation θ_M^* with minimal distance in comparison to predicted orientation:

$$\theta_M^* = \arg\min_{d_j}\left[\left|\bar{\theta} - \theta_j^{MLE}\right|\right], j = 0,\dots J , \qquad (16)$$

where J is the number of orientations with equal minimal or maximal matching scores (depend on type of histogram measurements).

Updates of the θ coordinate are as follows:

$$\hat{\theta}(k) = \bar{\theta}(k) + K_3 * \left(\theta_M(k) - \bar{\theta}(k)\right) , \qquad (17)$$

where $0 < K_3 < 1$ is a coefficient and θ_M is orientation value from angle histogram matching procedure.

For a given number of the available robot poses N, the number *of association hypothesis* N_{AH} is defined as:

$$N_{AH}(k) = \left\{N \mid N_T(k) \mid N_C\right\} = \sum_{N_T=1}^{\min(N_T(k))} N_{AH}(k \mid N_C) , \qquad (18)$$

where chosen subset of tracks $N_T(k)$ elements from N and possible associations of number of clutter measurements $(1 \le N_C \le N)$ and $N_T(k)$ dictate the total number of association hypothesis $N_{AH}(k)$. In most practical applications the number of association hypothesis $N_{AH}(k)$ is unknown and vary with time k. The pair $\psi=(N_T(k),N_C)$ presents *optimal assignment solution* [7].

An update phase of global localization process is presented in Fig. 6 for only one time step k. Fig. 6. presents modify Multiple Model Estimator (MME), which include N_T filters (N_T – number of tracks) for updating coordinates and probabilities in time step k. As said above, number of filters $N_T(k)$ varies with time k, i.e. so the overall estimator is *dynamic*. In this way, the number of association hypothesis N_{AH} varies with time, too.

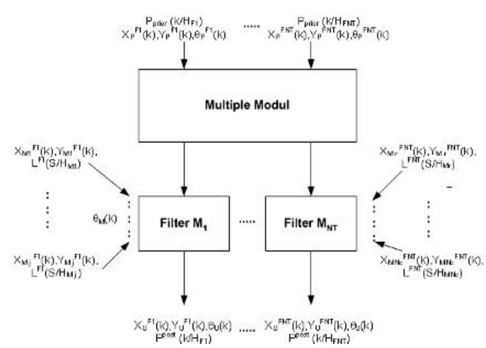


Fig. 6. *The modified MM estimator for one time step k.*

After the filters are initialized $N_T(0) = N$, they run recursively using the previous combined estimate. A Bayesian framework is used: starting with prior probabilities of each model being correct (i.e. the system is in a particular mode), the corresponding posterior probabilities are obtained. The likelihood functions are used to update the mode probabilities. In other words, from $N_T(k)$ hypotheses, a single hypothesis with maximal posterior probability at the end of each cycle is obtained at current position of mobile robot.

V. Probability Distribution System

A. Bayesian Framework

A probabilistic approach is required for localization as the observation of the features is a highly uncertain process and it depends upon real sensor and signal processing.

Definition of a *prior distribution* over the association hypotheses is given in the next form:

$$\tilde{p}(k \mid H_j) = \frac{L(H_j \mid H_{j*})}{\sum_{q=1}^{N_C} L(H_q \mid H_{q*})} . \tag{19}$$

The distribution of interest for tracking is the *posterior distribution*, also known as the *filtering distribution*, which holds all the observation up the current time step:

$$\widehat{p}(k \mid H_j) = \frac{L(S \mid H_j)\tilde{p}(k \mid H_j)}{\sum_{q=1}^{N_C} L(S \mid H_q)\tilde{p}(k \mid H_q)} . \tag{20}$$

Namely, when the robot moves, the whole distribution gets shifted according to the observed robot motion and the added uncertainty due to odometer drift. The result is a new form of probability distribution, which improves the robot's location model based on all of the available information.

The maximization of the posterior probability function density (MAP) follows:

$$\hat{p}^{MAP}(P) = \arg\max_{p} \left[\widehat{p}\left(k \mid H_j\right) \right] . \tag{21}$$

B. Reinitialization

This localization system allows the robot to localize itself with no knowledge of its initial position. It remain efficient even when the mobile robot is passively transported (lost problem) or by reinitialization process. When result of matching process between current scan and nodes in gate is

$$L\left(S \mid H_i\right) \le L_{treshold} . \tag{22}$$

the reinitialization process can be started.

VI. Experimental results

Described localization method is tested using a Pioneer 2DX mobile robot manufactured by ActivMedia Robotics equipped with an AMD K6-2 processor, 400MHz, 128 MB RAM, onboard computer manufactured by Versa Logic.
The specific simulation experiment is carried to demonstrate this global localization capacity. Start point was (23215, 10000), Robot goal point (35000, 7200) and start heading angle of mobile robot 0 (º). Real mobile robot final pose was measured at the end of the experiment. Fig. 7 presents a part of the environment used for this experiment with a denoted path.

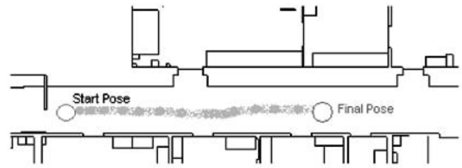


Fig. 7. Environment map with presentation of this experiment path.

Reinitialization process is made for example each 10 steps, which is represented on Fig. 8.

Fig. 8 (subfigures a)-c)) presents Pose 1, 4 and 10 and (subfigures d)-f)) Pose 11, 17 and 20 in range Global Localization using SG and modify MM approach with Probability Scalar Transform for Pose Correction and Obstacle Vector Transform for Orientation Correction. Each scan is made every 0.5 m of mobile robot moving. Probability distribution for pose 11 (Fig. 8.d)) presents 7 best results, when reinitialization is made. Further 7 poses contain probability distribution for 7 tracks (for example Fig. 8.e)) and then only for one track (Fig. 8.f)).

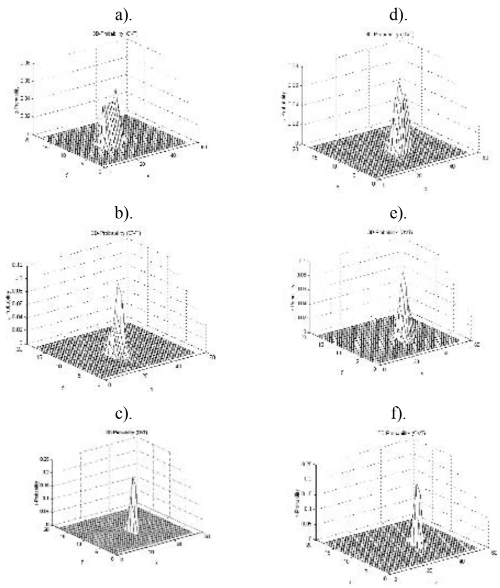


Fig. 8. Probability distribution of mobile robot during localization simulation experiment, by re-initialization.

Fig. 9 presents position results of PST+OVT based global localization process of mobile robot for this specific simulation experiment. Fig. 10 presents obtained results regarding orientation tracking with calibrated odometry and with proposed localization algorithm. In moment of reinitialization, more time is needed, because of computational complexity. It is possible to see it on Fig. 10, where orientation localization result is presented. Namely, in time step 11, 12, 22, and 23, the robot needs double time to continue global localization process. But the continuous localization process is adapted behind 1m of the traveled distance.

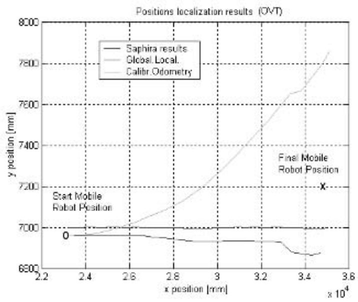


Fig. 9. Position localization results.

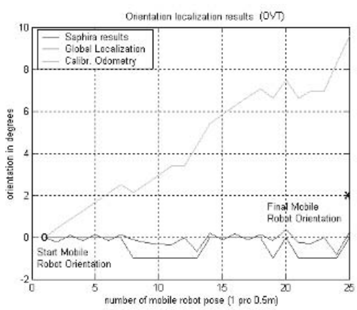


Fig. 10. Orientation localization results.

It is necessary to note that results of re-initialization depend of calibrated odometry, because mobile robot position and orientation are predicted using calibrated odometry. As a consequence, the reinitialization process becomes inefficient when for instance a wrongly estimated position is used as position for scan making. Similar result is in case of relocalization (lost experiment). The solution for this is recalibration process [11].

VII. Conclusion

The Soft-reduced Hypotheses Tracking (SRHT) localization method using soft-gating and modified multiple model approach with Probability Scalar Transform for pose correction and Obstacle Vector Transform for orientation correction is proposed. The method is suitable for real-time localization, as its computational complexity does not depend on the environment size. This is achieved by restricting the number of tracked location hypotheses. Experimental results confirm the validity of the proposed localization method.

References

[1] D. Lee, W. Chung, M. Kim.: “Probabilistic Localization of the Service Robot by Map Matching Algorithm”, in *Proc. Of International Conference on Control, Automation and Systems (ICCAS’2002)*, Korea, 2002, pp.1667-1627.

[2] S. Thrun, W. Burgard, D. Fox, “A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping”, *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA-2000), IEEE PRESS*, pp. 321-328, 2000.

[3] R. Karlsson: “Simulation Based Methods for Target Tracking”, Ph.D. Dissertation, Department of Electrical Engineering, Linkoeping University, Sweden, 2002.

[4] M. Hadzagic, H. Michalska, A. Jouan. “IMM-JVC and IMM-JPDA for closely manoeuvring targets”, the 35th Asilomar Conference on Signals, Systems and Computers, 2001, pp. 1278-1282

[5] Y. Bar-Shalom, X. R. Li, T. Kirubarajan: “*Estimation with Applications to Tracking and Navigation*”, A Wiley –Interscience Publication John Wiley& Sons Inc., 2001.

[6] T. Duckett, *Concurrent map building and self-localization for mobile robot navigation*, PhD Thesis, University of Manchester, 2000.

[7] L. Banjanović-Mehmedović, *Autonomus Mobile Robots Localization in large indoor Environments by using Ultrasound Range Sensors*, PhD Thesis, University in Zagreb, 2006.

[8] A. Elfes, “Using Occupancy Grids for Mobile Robot Perception and Navigation”, *Proceedings of IEEE International Conference on Robotics and Automation, Vol. 2*, pp. 727-733, 1988.

[9] R. Hinkel, T. Knieriemen, “Environment perception with a laser radar in a fast moving robot”, *Proceed. of the Symposium on Robot Control (SYROCO’88)*, Germany, 1988.

[10] J. Borenstein, Y. Koren: “The Vector field Histogram - Fast obstacle avoidance for mobile robots”, *IEEE Journal of Robotics and Automation*, Vol 7, No 3,1991, pp. 278-288.

[11] E. Ivanjko, I. Petrović, N. Perić, “An approach to odometry calibration of differential drive mobile robots”, *Proceedings of Inter*national Conference on Electrical Drives and Power Electronics EDPE'03, September 24-26, 2003, High Tatras, Slovakia, pp. 519-523.

Effective Arabic Character Recognition using Support Vector Machines

Mehmmood Abdulla Abd
Ajman University of Science and Technology
Faculty of Computer Science and Engineering
UAE
mahmood_abdullah@hotmail.com

George Paschos
N^th Research
Athens, Greece
gpaschos@netscape.net

Abstract - This paper proposes an Arabic character recognition system. The system focuses on employing Support Vector Machines (SVMs) as a promising pattern recognition tool. In addition to applying SVM classification which is a novel feature in arabic character recognition systems, the problem of dots and holes is solved in a completely different way from the ones previously employed. The proposed system proceeds in several phases. The first phase involves image acquisition and character extraction, the second phase performs image binarization where a character image is converted into white with black background, while the next phase involves smoothing and noise removal. In the fourth phase a thinning algorithm is used to thin the character body. The fifth phase involves feature extraction where statistical features, such as moment invariants, and structural features, such as number and positions of dots and number of holes, are extracted. Finally, the classification phase takes place using SVMs, by applying a one-against-all technique to classify 58 Arabic character shapes. The proposed system has been tested using different sets of characters, achieving a nearly 99% recognition rate.

I. INTRODUCTION

Offline Arabic Character Recognition (OACR) is a challenging problem; systems that address it will have contributed to the improvement of the computerization process [1]. Many scientists have intensively and extensively researched OACR of both printed and handwritten characters. Over the last forty years a great amount of research work in character recognition has been performed for Latin, Hindi and Chinese. The Arabic language serves as a script for several languages in the Middle East, Africa and Asia such as Arabic, Farsi, Urdu, Uygur, Jawi, Pishtu, Ottoman, Kashmiri, Old Hausa, Baluchi, Berber, Dargwa, Ingush, Kazakh, Kirghiz, Sindhi, and others. Moreover, all Muslims can read Arabic script as it is the language of AL-Quran. Despite these facts, research work on Arabic character recognition has not received much attention either because of its difficulties or due to lack of support in terms of funding and other utilities such as Arabic text databases, dictionaries, etc., and of course because of the cursive nature of its writing rules.

The cursive nature of the Arabic script makes the recognition of Arabic distinct from the recognition of Latin or Chinese scripts. In addition, most Arabic characters have from two to four different shapes/forms depending on their position in the word. Arabic writing has different font types. The font styles make Arabic character recognition hard and development of a system that is able to recognize all font styles is difficult. These styles encompass Ruq'a, Nastaliq, Diwani, Royal Diwani, Rayhani, Thuluth, Kufi and Naskh.

Arabic is cursively written from right to left (in both printed and handwritten forms) and the words are separated by spaces. It has 28 characters and each character has two or four different forms/shapes, depending on its position in the word, which increases the number of classes from 28 to 120. However, Arabic has no small or capital letters. The Arabic characters are connected in the word on the base line. Furthermore, some characters in the Arabic language are connectable from right only, these are**:** ا,د,ذ,ر,ز,و. Some of the right-connectable characters cause an overlapping between subwords, for instance "Waow و ". Overlapping can be addressed with a contour-following algorithm [2]. Therefore, such characters divide the words into two subwords, when they appear in a word.

Some characters have exactly the same shape and some diacritics that make them differ from each other. These diacritics involve a dot, a group of dots, or a zigzag (hamza). The presence or absence of diacritics has a very important effect on Arabic word meaning. For instance, the word "حُب " means, "love " and "حَب " means "grain ",

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 7–11.

where the meaning completely depends on the diacritics. Diacritics may appear above or below the base line (letter). Some Arabic characters have one to two holes within the character's body. The dot is also another feature that is used to distinguish among similar characters. The maximum number of dots that may appear above the character is three and below the character is two. A thinning algorithm may effectively deal with them.

Arabic character recognition falls into either online or off-line category, each having its own recognition algorithms and hardware. This paper deals with isolated offline Arabic character recognition. The purpose of the proposed work in this paper is to build a high-accuracy Arabic character recognition system using improved feature extraction and optimized Support Vector Machines (SVMs). The objectives of this research are to a) improve the recognition rate in Arabic character recognition, b) improve the performance of SVMs. The proposed methodology is described by the following processing phases:

1. Image acquisition and character extraction
2. Image binarization
3. Smoothing and noise removal
4. Character thinning
5. Feature extraction
6. Classification using multi-class SVMs.

The remaining of the paper is organized as follows. Section II describes preprocessing, section III presents the feature extraction methodology, section IV presents the multi-class SVM classification system followed by the results obtained in section V, while section VI provides a recapitulation and suggestions for future work.

II. PREPROCESSING

The Arabic character features are extracted from gray-level image, which is scanned by a regular flat scanner. The threshold value is chosen based on trial and error. This threshold value is utilized to yield a white character body with a black background. Then, the character's body is isolated from the image background. A binary image is cleaned up and introduced to the feature extraction phase. Mathematical morphology is utilized to remove noise and to smooth the character's body. It is worth mentioning that this technique has not been used in the Arabic character recognition techniques.

Two morphological operations that are mainly used are opening and closing. Closing fills small gaps in an image, which eliminates small holes in the image's contour, and opening opens small gaps or spaces between touched objects in the character's image. This is useful to break narrow isthmuses and eliminate small objects. Both operations employ the same basic morphology operations, which are dilation and erosion, using the same structural elements. Then, a sequential morphological thinning algorithm is used to remove spurious pixels from the edge of the character's body.

III. FEATURE EXTRACTION

The proposed system deals with isolated Arabic characters to recognize an unknown character by deciding to which class it belongs. After extracting all structural and statistical features of the Arabic characters, the feature vector is fed to the SVM classifier. These features consist of the first three moment invariants, the number and position of dots, the number of holes in the character body, and the number of pixels in the dot area, as described below.

A. Moment invariants

The Hu moment invariants are calculated from each character image as described in [3]. These moment invariants are insensitive to image translation, scaling and rotation, thus, they have the desired properties to be used as pattern descriptors.

The first three moment invariants are utilized to decrease the number of features, and consequently speed up training and classification, where the absolute value of the logarithm of each moment invariant is in fact computed instead of the moment invariants themselves. Using the logarithm reduces the dynamic range, and absolute values are taken to avoid dealing with the complex numbers, which may result when computing the negative values of log of moment invariants [3]. The invariance of moments is important and not their signs, therefore absolute values are used.

B. Number of dots and their position

The number of dots and their positions play important roles in Arabic character recognition. Some Arabic characters have the same shape but the number of dots and their positions make them differ from each other. For instance, Ta " ت " and Tha "ث " have the same shape and they differ in the number of dots, and Noon" ن " and Ba " ب "they differ in their dot positions. Consequently,

using these features as another descriptor of Arabic characters may increase recognition accuracy.
The coordinates of the four corners of the main body of characters are used to determine the position of the dots: a) the label matrices of image objects are computed based on 8-connectivity neighborhood (the label matrix of an object is a vector of labeled pixel coordinates [3]); b) then, the four coordinates of the bounding box are computed to determine the position of the dot relative to the main character body. It is worth mentioning that this technique has not been used in Arabic character recognition systems.

C. Number of holes

Holes is another structural feature of Arabic characters. Some Arabic characters have either one or more holes. For instance, Sad " ص " has one hole while Middle-Ha " ـهـ " has two holes and End Ha " ـه " has one hole. The maximum number of holes is two. The Euler number was utilized to find the number of holes in an Arabic character in this study. The Euler number is a scalar equal to the number of objects in the image minus the number of holes in those objects [3]. The Euler number is computed based on the number of objects (NOB) in the image of a character as extracted by using label matrices. It is worth mentioning that this technique has not been used in Arabic character recognition systems.

IV. MULTI-CLASS SVM CLASSIFICATION

SVMs are basically binary classifiers and it is not straightforward to turn them into multi-class (N-class) recognition systems. There are several methods invented to a construct multi–class SVM. The most typical such method is to construct N SVMs, each of which classifies one class against all the other classes. This method is commonly called one-against-all (1-v-all) [5]. The second method that can be used is to combine all possible two-class classifiers, where for an N-class problem N(N-1)/2 classifiers must be constructed. This is commonly referred to as one-against-one (1-v-1). Decision Directed Acyclic Graph (DDAG) and Max Win Algorithm (MWA) are also used to construct multi-class SVMs based on (1-v-1). Further details can be found in [6],[5].

The one-against-all method is used in this study. N SVMs are constructed, where *N* is the number of classes. For $i = 1,2,..,N$, the i[th] SVM is trained with all the samples in the i[th] class considered as positive examples and the samples from all other classes considered as negative examples. Given $\{(x_1, y_1), (x_2, y_2),\ldots,(x_k, y_k)\}$ as the training data set, where $x_j \in R^n$ are the n-dimensional samples (feature vectors) and $y_j \in \{1,2,\ldots,N\}$ are the corresponding class labels (j=1,2,…,k), the i[th] SVM solves the following problem [6]:

$$\begin{aligned}&\min_{w^i,b^i,\xi^i} \left\{\frac{1}{2}\left(w^i\right)^T w^i + C\sum_{j=1}^{k}\xi_j^i\right\}\\ &\left(w^i\right)^T \Phi\left(x_j\right) + b^i \geq 1-\xi_j^i,\ \text{if } y_j = i\\ &\left(w^i\right)^T \Phi\left(x_j\right) + b^i \geq 1+\xi_j^i,\ \text{if } y_j \neq i\\ &\xi_j^i \geq 0,\ j = 1,\ldots\ldots k\end{aligned} \tag{1}$$

where $\frac{1}{2}\left(w^i\right)^T w^i$ is the regularization term (objective function), $C\sum_{j=1}^{k}\xi_j^i$ is a cost term (training error) used to constrain the solution in cases of non-separable data (ξ are 'slack' variables introduced for this purpose [5]), function Φ is used to map the training data into a higher dimensional space, and C is a penalty factor set by the user.
The above equation will be solved for each class $i = 1,2,..,N$, where the attempt is made to balance between the regularization term and the training errors. Thus, there are N decision functions

$$\begin{aligned}&\left(w^1\right)^T \Phi(x) + b^1\\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\ &\left(w^N\right)^T \Phi(x) + b^N\end{aligned} \tag{2}$$

and x is said to belong to the class which gives the largest value for the decision function, i.e.,

$$\text{class of } x \equiv \arg\max_{i=1,\ldots,N}\left(\left(w^i\right)^T \Phi(x) + b^i\right) \tag{3}$$

In practice the dual problem, formulated in terms of Lagrange multipliers, is solved, i.e., maximize

$$\begin{aligned}&\sum_i a_i - \frac{1}{2}\sum_{i,j} a_i a_j y_i y_j \Phi(x_i)\cdot\Phi(x_j)\\ &0 \leq a_i \leq C\\ &\sum_i a_i y_i\end{aligned} \tag{4}$$

which has the following solution:

$$w = \sum_{i=1}^{N_S} a_i y_i \Phi(x_i) \tag{5}$$

where N_S is the number of support vectors [5].

Replacing the inner product between Φ in (4) with a function simplifies things, wherein kernel functions are introduced. In this work, Gaussian Radial Basis Functions (GRBF) are employed as kernel functions:

$$K(x_i, x_j) = e^{\left(-\gamma \| x_i - y_j \|^2\right)}, \gamma > 0 \qquad (6)$$

The linear kernel is a special case of GRBF as Kreethi and Lin have shown in [8]. Sigmoid kernels produce results generally comparable to GRBF kernels. However, GRBF has less numerical complexities in comparison to other kernels [7][6]. GRBF has a control parameter, γ, which along with the cost factor C are the two tunable parameters in the SVM optimization problem. As suggested in [9], a useful way to determine these is to perform grid-search with cross-validation. A two-stage procedure can also be followed, where after identifying a good region in the (C, γ) space, a finer search can be conducted which may improve the results somewhat in some cases. The best parameters thus determined are used in the classification.

V. RESULTS

Figure 1 shows a screen shot of a typical run of the SVM classification system. The software used is LIBSVM [10] with additional utilities developed in Java. The set of samples obtained from 58 characters/shapes is divided into two to be considered as training and testing set, respectively. In the final testing stage, ten-fold cross validation is used to determine good C and gamma parameter values, followed by actual classification using these values.

The recognition rates obtained are in the range of 98-99% correct classification, depending on the kind and number of classes (characters) used. Comparatively, with the same number of character classes (58), [11] achieved a recognition rate of 91%.

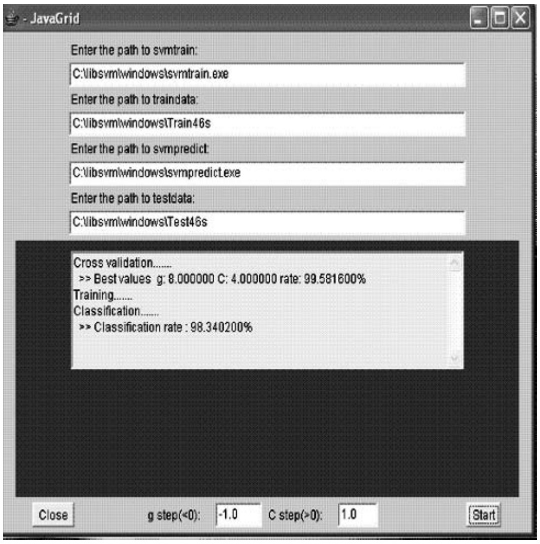


Fig. 1. Screen Shot of the SVM classification using 58 different classes (character forms): a 98.34% recognition rate is shown.

1. VI. CONCLUSIONS

Employing SVMs as a recognition tool in Arabic character recognition system has shown promising results that produce high recognition rates. The proposed system relies on multi-class SVM classification and moment invariant features. This system can be applied on any pattern such as fingerprints, iris and characters/letters. This system can further be used for multi-fonts recognition for any languages. A comparison of the proposed system with equivalent neural network architectures is to be performed.

REFERENCES

[1] Nazif A., A System For "The Recognition Of The Printed Arabic Characters", Master's thesis, Faculty of Engineering, Cairo University, (1975).
[2] Amin A.," Recognition Of Printed Arabic Text Based On Global Features And Decision Tree Learning Techniques", Pattern Recognition Letters, 33 (8), 2003, pp. 1309-1323.
[3] C.Gonzalez Rafael, E.Woods Richard and L.Eddins Steven, Digital Image Processing using MATLAB, Pearson Prentice Hall, USA, 2004.
[4] Michio S., Hiroshi F. and Gisaku N.,"A Thinning Algorithm For Digital Figures Of Characters", 4th IEEE Southwest Symposium on Image Analysis and Interpretation, Austin, Texas, pp. 83-88, 2000.
[5] Christopher J.C., "A Tutorial on Support Vector Machines for Pattern Recognition", http://aya.technion.ac.il/karniel/CMCC/SVM-tutorial.pdf, 2005.
[6] Chih-Wei Hsu and Chih-Jen Lin, "A Comparison Of Methods For Multi-Class Support Vector Machines", IEEE Transactions on Neural Networks, Vol. 13, No. 2, 2002.

[7] Eugene A. Borovikov, "An Evaluation of Support Vector Machines as a Pattern Recognition Tool", http://www.umiacs.umd.edu/users/yab/SVMForPatternRecognition/report.pdf, University of Maryland at College Park, 2005.
[8] Kreethi, S. S. and Lin C. J., "Asymptotic Behaviors Of Support Vector Machines With Gaussian Kernel", Neural Computation, 15(7), pp. 1667-1689, 2003.
[9] C-W. Hsu, C-C. Chang, and C-J. Lin, "A Practical Guide to Support Vector Classification", www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.
[10] C-C. Chang and C-J. Lin, "LIBSVM – A Library for Support Vector Machines", www.csie.ntu.edu.tw/~cjlin/libsvm
[11] Hosseini H. M. and B. Abdesselam, "A System For Arabic Character Recognition", in Proceedings of the 2nd International Conference on Intelligent Information Systems, pp: 120-124, 1994.

This page intentionally blank

Extracted Structural Features for Image Comparison

Pornchai Mongkolnam[a], Thanee Dechsakulthorn and Chakarida Nukoolkit[a]
[a]School of Information Technology
[a]King Mongkut's University of Technology Thonburi
Bangkok, Thailand

***Abstract*- We present a method that extracts structural features of images. The method is based on both a region-based analysis and a contour-based analysis. The image is first segmented, based on its pixels' information. Color information of each segmented region is performed by using the hue-saturation-value color space. Area of each region is also extracted by counting the number of bound pixels. Location of each region is computed as a center of the region's convex hull. A contour of the region is approximated by a B-spline approximation to obtain its control polygon and curve in the limit. The region's convex hull is obtained from the control polygon. For multi-scale features, we apply Chaikin's algorithm to the control polygon for finer level of control polygons, which could be used in a coarse to fine comparison. Curvature information of the B-spline curve fitting could also be used in the comparison. Our method could be used in many interesting applications including image retrieval, image classification, image clustering, image manipulation, image understanding, pattern recognition, and machine vision.**

I. Introduction

There is an ever-increasing need for a simple, yet effective and efficient way to analyze, retrieve, cluster, explore, and display digital mages and videos. Similar need for a large document collection is also desired [1, 2]. The most popular image search Web sites such as Yahoo! and Google are the irrefutable evidence. To improve the search results various techniques have been invented in order to incorporate other relevant features such as shape, color, and texture into a mere text-based image search. Keyword-only search has some drawbacks in that keywords are context dependent and do not allow for unanticipated search. Furthermore, language barriers and lacks of uniform textual descriptions make them ineffective.

Several existing work uses B-spline curves to represent profile shapes of 3D objects [3, 4]. A contour of the object is first extracted and then approximated by the B-spline curve, which is in turn used for curve matching in a data retrieval application. The object matching is an integral part for many applications of shape modeling, machine vision, and image processing. The B-spline curves and their curvatures are widely and effectively used for curve representation of the object contours instead of the far higher degree Bezier curves because they possess some very attractive properties such as smoothness, compactness, local shape controllability, and affine transformation invariance. In addition to using B-spline approximation, the Chaikin's algorithm [5] is used to refine the matching curves at many different scales in this work. For images, little work has been done on applying the B-spline concept to structurally represent the images' components. Some work merely allows user to sketch an outline as well as specify color information before submitting the query to the search engine [6]. Therefore, this work has extended the already existing 3D object's curve concepts to 2D images in order to help represent structural shapes or component-level contents within the images. Knowing the shapes, components and their spatial layout relationships would certainly yield a good comparison in image databases, comparatively to understanding a molecular structure of an element. Our work would definitely find a useful place in various important fields such as machine vision, pattern recognition, image retrieval, clustering, exploration, manipulation, understanding and visualization. A good overview of using shapes for content-based image retrieval (CBIR) can be found in [7, 8, 9].

The paper is organized as follows. First, we present overview and related work on B-splines, curve fitting or curve approximation, Chaikin's algorithm, image shape modeling, and content-based image retrieval. Thereafter our work and its results on extracting segmented regions' features are discussed. Last, a conclusion and future work on extending the image shape representation to image retrieval, clustering, exploration, and visualization is given.

II. Related Work

This work has been built on some prior work in a 2D and 3D shape matching and curve representation that apply B-spline and its curvature to approximate the image or object's profile. Over the past thirty years work on shape has been an active research area and was mainly driven by object recognition. One of the recent work [4] proposes a novel 2D shape-matching algorithm based on the B-spline curves and its Curvature Scale Space (CSS) image. The CSS image [10] is robust with respect to noise and affine transformation and is chosen as a shape representation for a digital planar curve. The representation is computed by convolving the curve with a Gaussian function at different scaling levels. The CSS is suitable because the B-splines have advantages of being continuous curve representation and affine invariant. The algorithm first smoothens the B-spline curve of an input shape and constructs the CSS image. It then extracts the maxima of CSS image and performs matching.

Due to the B-splines' attractive properties, Reference [3] chooses the B-splines for curve modeling in matching 2D objects such as aircrafts and handwriting over other approaches such as the Fourier descriptors, the polygonal approximation, the medial axis transform, the moments, and

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 13–18.

the curvature invariant. Their algorithm attempts to match and classify planar curves that are modeled as B-splines, independent of any affine transformations. Two methods are presented. First, the control points of the prototype curves are globally related to the knot points of a given sample curve and then are compared. Second, a sum of the residual error between each prototype curve and the given sample curve is compared.

A new image database retrieval method based on shape information and deformable template matching process is proposed by using the following two features to represent shape of an image: a histogram of the edge directions and the invariant moments [11]. Euclidean distance between the edge direction histograms is used as a matching score. The shape of the image is also represented in terms of the seven second-order and third order invariant moments. A region-based image retrieval method that employs a new image segmentation technique using circular filters based on Bayes' theorem and image texture distribution is proposed by Reference [12]. After segmentation, extracted features of each region including color, texture, normalized area, shape, and location are recorded and compared against other images.

Reference [13] proposes shape retrieval from image databases, which are composed of boundary contours; the method is based on indexing structural features in terms of convex/concave parts and quantized directional features along the contours. The method exploits the feature transformation rules, which is obtained by an analysis of some particular types of shape deformations, to generate features that could be extracted from deformed patterns. Most work previously mentioned is performed on a single scale shape analysis, which is not able to provide a robust representation. A multi-scale analysis for shapes [14] is used to derive a hierarchical shape representation in that the shape details are progressively screened out whereas the shape characterizing elements are preserved. Using the graph structures representing shape parts at different scales, the coarse-to-fine matching could be performed.

Besides using a curve matching, a shape matching can also be achieved by matching skeletal graphs (medial axis graphs) [15, 16]. The medial axis has been used for matching shapes because outline curves do not meaningfully represent the interior of the shapes. The shock graph, which is the medial axis endowed with geometric and dynamics information, is used because it gives a richer description of shapes. In summary, there are various existing techniques being used on shapes in image processing as shown in Fig. 1. A complete overview could be found in [17].

III. OUR METHOD

Our method begins with image segmentation in order to globally identify structural components of the image by applying the JSEG algorithm [18, 19]. It involves two independent steps: color quantization and spatial segmentation. First, the image pixel colors are quantized to several classes called class maps, which are used to differentiate regions in the image. Second, a region growing

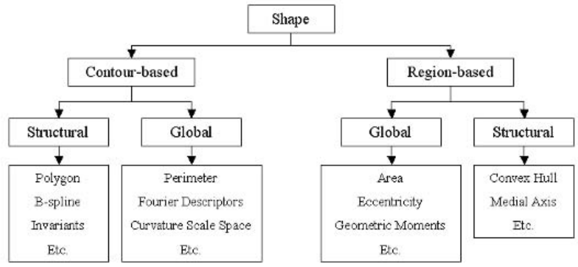


Fig. 1. Classification of shape representation and descriptor techniques.

method is used to segment the image based on the property of the obtained class maps. A good segmentation method would certainly help obtain good image contours and segments along with their relationships, which significantly impact on quality of subsequent work in image retrieval, clustering, and exploration.

After the segmentation, a boundary of each extracted segment is approximated by a B-spline curve. In general, a B-spline curve [20, 21] is more widely used to represent a complex curve than its far higher degree Bezier curve counterpart because of its local control property and ability to interpolate or approximate a curve with lower degree. The B-spline curve is a generalization of the Bezier curve and has more desired properties than the Bezier curves. The B-spline curve is generated from its control points and contained in the convex hull of its control polyline or polygon. An affine transformation such as rotation, translation, and scaling can be applied to the B-spline control points quite easily instead of to the curve itself. This results in an affine invariance property, where manipulation can be done to the control points instead of to the curve itself. Therefore, speed could be improved when a curve matching is done.

The B-spline curve, *C(u)*, is defined as:

$$C(u) = \sum_{i=0}^{h} N_{i,p}(u) P_i$$

where P_i is a control point, p is a degree, u is parameter, and $N_{i,p}$ is a B-spline basis function and is defined as:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

where u_i is known as a knot, a corresponding place where two curve segments join with certain continuity. Fig. 2 illustrates a cubic B-spline curve generated from its eight control points. The control points are shown in dark circles. Lines connecting control points are a polyline, and they capture the overall shape of the curve.

Fig. 2. A cubic B-spline curve with 8 control points shown in dark circles.

Given data points, the B-spline curve can either interpolate or approximate those points. Interpolation is not practical here because of a large number of data points from the image contours. With such number, the resulting curve could be wiggly and with a lot of control points. Therefore, the B-spline curve approximation is a preferred choice for our work. In the approximation the B-spline curve does not have to pass through all data points except the first and last data points. The number of the B-spline control points would reflect the goodness of the approximation. For each data point, an error distance is computed as the square distance between the data point and a corresponding point on the B-spline curve. A sum of all square error distances is used to measure how well the B-spline curve approximates the data points. An objective is to minimize the sum of the error distance in order to get a good approximation of the data points.

A problem statement of the B-spline approximation is posed as:

Input: Given a set of n+1 ordered data points, $D_0, \dots, D_n$.

Output: A B-spline curve of degree p with h+1 control points, $P_0, \dots, P_h$, which satisfies the following two conditions.

- The curve interpolates the first and last data points, D_0 and D_n and
- The curve approximates the data points in the sense of a least square error distance.

Since the curve contains the first and last data points, we would have $D_0 = P_0$ and $D_n = P_h$. The curve equation is now written as:

$$C'(u) = N_{0,p}(u)D_0 + \left(\sum_{i=1}^{h-1} N_{i,p}(u)P_i\right) + N_{h,p}(u)D_n$$

Let parameters be $t_0, \dots, t_n$. The number of parameters is equal to the number of the data points because we want to find the corresponding point on the curve for each data point. The centripetal parametrization is used and computed as:

$$\frac{\Delta_i}{\Delta_{i+1}} = \left[\frac{\|\Delta x_i\|}{\|\Delta x_{i+1}\|}\right]^{1/2}$$

where $\Delta_i = t_{i+1} - t_i$ and $\Delta x_i = D_{i+1} - D_i$.

The sum of all square error distances is computed as:

$$f(P_1, \dots, P_{h-1}) = \sum_{k=1}^{n-1} |D_k - C(t_k)|^2$$

The control points, $P_1, \dots, P_{h-1}$, are solved such that the objective function $f(.)$ is minimized.

Most of segmented contours are closed and hence resulting in the control polygons. In a multi-scale, coarse-to-fine shape representation, each control polygon is used to compute its corresponding coarser polygon, called a convex hull [22]. There are numerous applications for convex hulls such as hidden object determination, collision avoidance, and shape analysis. Reference [23] uses convex hull shapes along with concavity features of regions for partitioning search space in medical image databases in order to speed up a search and retrieval time. We apply the fast Andrew's monotone Chain algorithm to control point set of the control polygon, obtaining the convex hull.

In addition to using the control polygon for getting the convex hull, it would be used as a first-level shape representation of the B-spline approximation curve. The Chaikin's algorithm is recursively applied to the first-level control polygon for subsequent finer level shape representations. In the limit, the resulting control polygon is simply the B-spline curve. The Chaikin's algorithm has been used for a subdivision curve and could be elevated to a subdivision surface. It is defined as: given a control polygon, defined by $\{P_0, \dots, P_n\}$, a new refined sequence of control points is $\{Q_0, R_0, Q_1, R_1, \dots, Q_{n-1}, R_{n-1}\}$, where each new pair of points Q_i, R_i is computed to be at $\frac{1}{4}$ and $\frac{3}{4}$ of a line segment $\overline{P_i P_{i+1}}$. Fig. 3 illustrates resulting curves from a coarse-to-fine scale.

Other than using multi-scale curve as a primary structural feature, our method uses areas, hue-saturation-value (HSV) color information, spatial information and relationships of segmented regions. Two reasons to include area as feature are: region's significance and region's exclusion. By knowing a relative size of each region, we would exclude the less important regions and focus more on the significant ones. This would dramatically speed up a comparison and matching process. The comparison would be blindly done and successful to a certain extent if only the shape feature is used once a refine comparison is needed when shapes are similar. To lessen that problem, the HSV color histogram information is used for both with and without shape feature. Images retrieved by using a global color histogram may not be semantically related. Nevertheless, at a coarse level the color information tends to be more effective than the shape feature because of how human visual perception is toward chromaticity. Euclidean distance is used here to define the similarity of two color histogram representations.

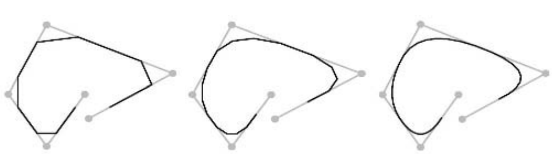


Fig. 3. Chaikin's algorithm applied to a control polyline three times, subsequently from left to right.

Locations and their relative proximities of segmented regions are used as spatial information to enhance comparison and matching as well. The image could be divided into 4x4 cells so that the extent of each segmented region is known. We also record the neighbors of each region and whether they are adjacent or enclaved.

IV. RESULTS

Fig. 4 illustrates the B-spline curve approximation to two test data sets: one is the open-ended, and the other is the closed-ended. We apply the curve approximation to a contour or boundary of each segmented region. An output of the curve approximation is a set of the curve's control points, or the so-called control polygon of the closed curve. Fig. 5(a) shows one of the control polygons of segmented regions. The control polygon is used either to generate the limit B-spline curve or to obtain a convex hull, as shown in Fig. 5(b). In a multi-scale representation, we would apply the Chaikin's algorithm repeatedly to the control polygon and arrive at a more refined control polygon. Fig. 5(c) shows a one-time application of the algorithm to the control polygon. Fig. 6 summarizes the whole process of our method, starting with a segmentation of an original input image, to B-spline curve approximation, to obtained control polygons, to corresponding convex hulls, and finally to the refined control polygons. Note that some small, segmented regions are considered too trivial to be meaningful and are to be rid of.

V. CONCLUSION

In this paper we have presented a technique for extracting structural shapes of images for uses in various image applications such as retrieval, clustering, and manipulation. Our focus was the multi-scale shape representation from a convex hull to control polygons at different scales and eventually to b-spline curves in the limit. We also propose the use of regional color and spatial information, and their relationships in addition to the hierarchical shape features.

We believe our method is unique from other methods in that the method exploits the information at the component level of the images. Furthermore, the B-spline approximation is used to arrive at the first level control polygons, which are in turn used to obtain the corresponding convex hulls. In order to speed up and enhance comparisons, the method applies the Chaikin's algorithm to the first level control polygons repeatedly to get finer control polygons. The comparison could begin from the convex hull level, to the first level control polygon, and to the subsequent control polygons as so desired.

VI. FUTURE WORK

Much future work remains to be done, especially the application of our work to image retrieval and clustering. We have demonstrated how the structural features could be extracted from the images. The multi-scale representations from the convex hull to finer level of control polygons as well as the limit curve could be effectively used for comparison. We plan to find an efficient way to index images by combining our hierarchical structural features and regional color and spatial information. We also look into how to apply wavelets to those extracted features because of their multi-resolution supporting nature. An image structural feature extraction could nicely be extended to 3D object segmentation and matching. The different-level control polygons would be extended to multi-level mesh control polyhedrons. The limit B-spline curve would then be B-spline tensor product (surfaces). The Chaikin's algorithm still works for 3D meshes as it does for the 2D planar curves.

Another interesting avenue for extension of this work is to have a better display, after obtaining resulting images from the image retrieval and clustering. Better ways to render meaningful results would help us better understand the results and their relationships, which otherwise would be too subtle to comprehend and interpret. Future work in visualization in both a two-dimensional and three-dimensional setting is needed on top of just a plain old grid-like display as seen in the Google or Yahoo! image search. To display the result in 2D we could apply the concept of pictorial summary [24]. The dominant images of the result or the cluster would be given more spaces and prominent location than the less significant ones. In 3D display, the Catmull-Clark subdivision surface [25] would be used together with the terrain modeling [1, 2] and pictorial summary. The results would be shown as a relief map of natural terrain where the dominant groups of images are shown at the taller peaks. Inherent multi-resolution properties of both the subdivision surface and pictorial summary could be used to render the results from coarse to fine details.

ACKNOWLEDGMENT

We would like to extend our appreciation for the help and support from the School of Information Technology (SIT) at King Mongkut's University of Technology Thonburi (KMUTT). This work is supported by the National Science and Technology Development Agency (NSTDA) of Thailand under grant F-31-205-20-01 and the KMUTT's Research and Intellectual Property Promotion Center (RIPPC). We would like to acknowledge Y. Deng and B.S. Manjunath, L. Vandevenne, and D. Sunday for their segmentation code, flood-fill code, and Chain hull code, respectively. We wish to thank J. Z. Wang for his image collection used in our work. Tutorial on B-splines by C. K. Shene is very informative and helpful.

REFERENCES

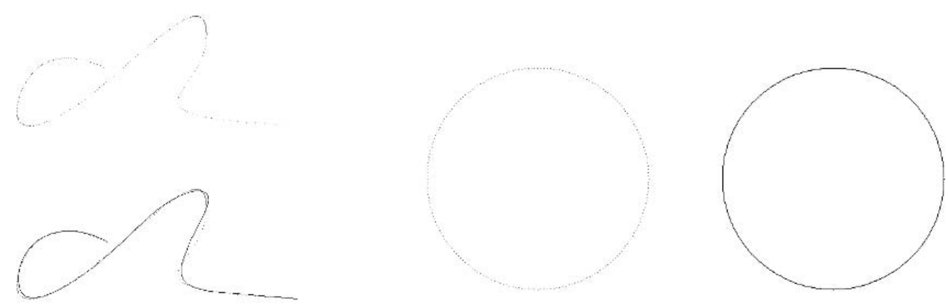
[1] N.E. Miller, P.C. Wong, M. Brewster, and H. Foote, "Topic Islands™—A wavelet-based text visualization system," *Visualization'98*, Oct. 1998, pp. 189-196.

[2] J.A. Wise, J.J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, "Visualizing the non-visual: spatial analysis and interaction with information from text documents," *Proceedings of IEEE'95 Information Visualization*, Oct. 1995, pp. 51-58.

[3] F.S. Cohen, Z. Huang, and Z. Yang, "Invariant matching and identification of curves using B-splines curve representation," *IEEE Transactions on Image Processing*, vol. 4, no. 1, Jan. 1995, pp. 1-10.

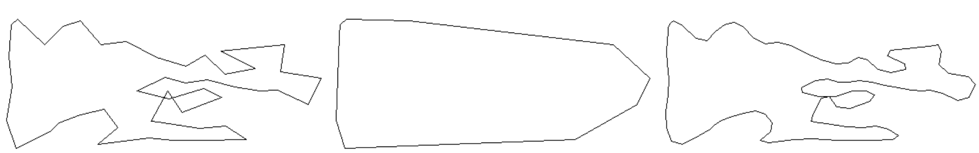
[4] Y. Wang and E.K. Teoh, "A novel 2D shape matching algorithm based on B-spline modeling," *International Conference on Image Processing, ICIP'04. 2004*, vol. 1, 24-27, Oct. 2004, pp. 409-412.

[5] G. Chaikin, "An algorithm for high speed curve generation," *Computer graphics and image processing*, pp. 346-349, 1974.
[6] A. Del Bimbo and P. Pala, "Visual image retrieval by elastic matching of user sketches," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, Feb. 1997, pp. 121-.132.
[7] A. Pentland, R.W. Picard, and S. Sclaroff, "Photobook: Tools for contents-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, no. 3, 1996, pp. 233-254.
[8] A.W.M Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, Dec. 2000, pp. 1349-1380.
[9] J.Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: semantics sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, Sep. 2001, pp. 947-963.
[10] F. Mokhtarian and A.K. Mackworth, "A theory of multi-scale, curvature-based shape representation for planar curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, Aug. 1992, pp. 789-805.
[11] A. Vailaya, Y. Zhong, and A.K. Jain, "A hierarchical system for efficient image retrieval," *Proceedings of the 13th International Conference on Pattern Recognition*, vol. 3, Aug. 1996, pp. 356-360.
[12] B.C. Ko and H. Byun, "FRIP: A region-based image retrieval tool using automatic image segmentation and stepwise Boolean AND matching," *IEEE Transactions on Multimedia*, vol. 7, no. 1, Feb. 2005, pp. 105-113.
[13] H. Nishida, "Shape retrieval from image databases through structural feature indexing," *Vision Interface '99*, Trois-Rivieres, Canada, May 1999, pp. 328-335.
[14] A. Del Bimbo and P. Pala, "Shape indexing by multi-scale representation," *Image and Vision Computing*, vol. 17, no. 3-4, 1999, pp. 245-261.
[15] T.B. Sebastian and B.B. Kimia, "Curves vs. skeletons in object recognition," *International Conference on Image Processing*, 2001. Proceedings, vol. 3, Oct. 2001, pp. 22-25.
[16] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson, "Skeleton based shape matching and retrieval," *Shape Modeling International*, May 2003, pp. 130-139.
[17] D. Zhang, "Image retrieval based on shape," *Ph. D. Dissertation*, Mar. 2002, Monash University, Australia.
[18] Y. Deng and B.S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI '01)*, vol. 23, no. 8, Aug. 2001, pp. 800-810.
[19] Y. Deng, B.S. Manjunath, and H. Shin, "Color image segmentation," *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR '99, Fort Collins, CO, vol. 2, Jun. 1999, pp. 446-451.
[20] G.E. Farin, "*Curves and Surfaces for CAGD: A Practical Guide*," Morgan Kaufmann, 5th edition, Oct. 2001.
[21] C.K. Shene, "Introduction to computing with geometry," *http://www.cs.mtu.edu/~shene*.
[22] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, "*Computational Geometry: Algorithms and Applications*," Springer, 2nd edition, 2000.
[23] N.M. Sirakov and P.A. Mlsna, "Search space partitioning using convex hull and concavity features for fast medical image retrieval," *IEEE International Symposium on Biomedical Imaging: Macro to Nano*, vol.1, Apr. 2004, pp. 796-799.
[24] M.M. Yeung and B-L. Yeo, "Video visualization for compact representation and fast browsing of pictorial content," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 5, Oct. 1997, pp. 771-785.
[25] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Computer-Aided Design (CAD)*, 1978, pp. 350-355.



(a) (b)

Fig. 4. B-spline curve approximation to the (a) open-ended and (b) closed-ended test data.



(a) (b) (c)

Fig. 5. (a) Example of one of the obtained control polygons, (b) its convex hull, (c) and its finer control polygon after applying a one-time Chaikin's algorithm to the control polygon.

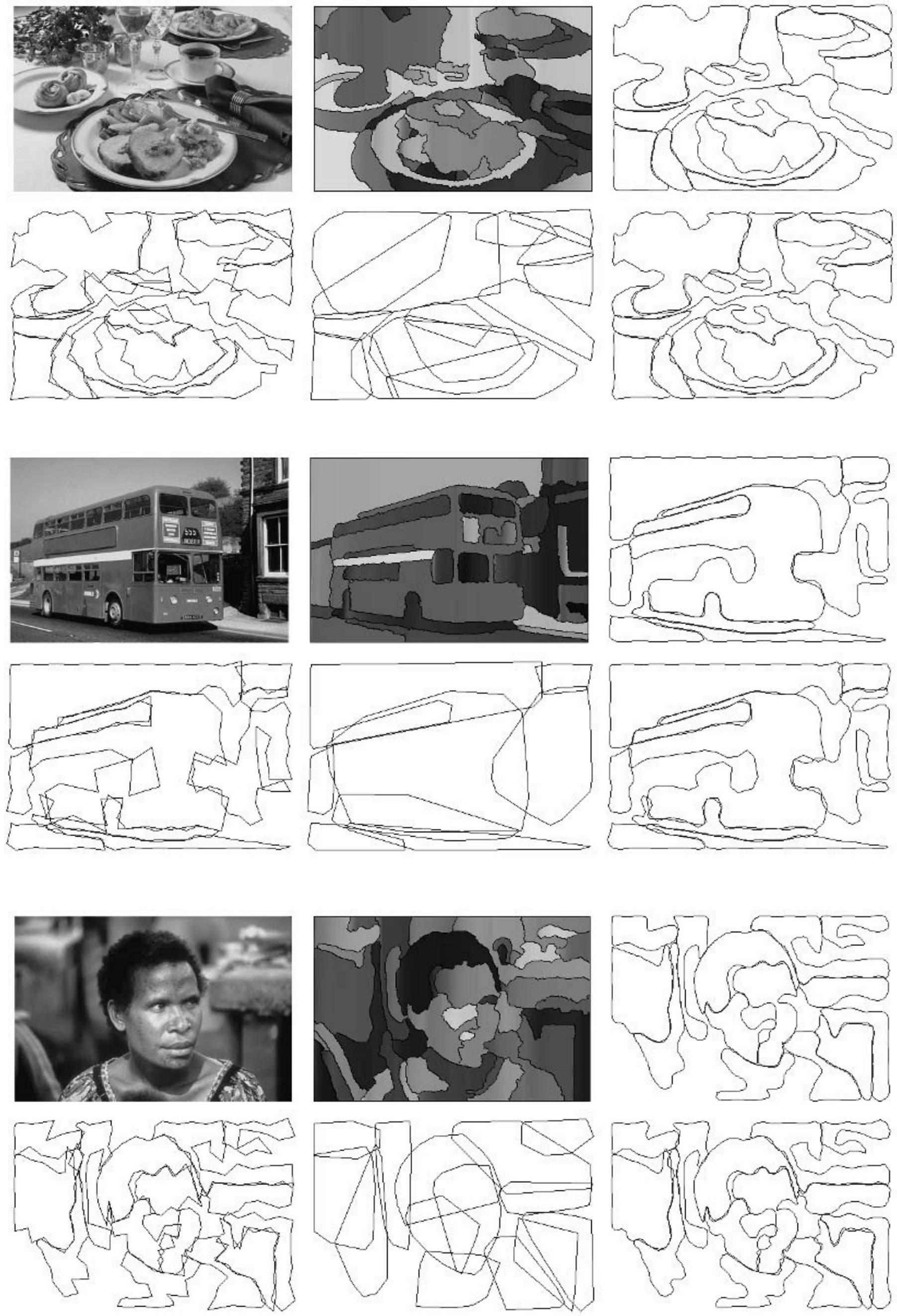


Fig. 6. Top left pictures are original images. Top middle pictures are segmented images. Top right pictures are B-spline approximation. Bottom left pictures show control polygons. Bottom middle pictures show corresponding convex hulls. And bottom right pictures show the one-time refined control polygons after applying Chaikin's algorithm to the control polygons.

The Functional Model of a Robot System Which Presents a Visual Servoing Control

R.C. Tarca, I.C. Tarca, A. Tripe-Vidican, T. Vesselenyi
University of Oradea, Romania
e-mail: rtarca@uoradea.ro, nelut@uoradea.ro, atripe@uoradea.ro, tvesselenyi@yahoo.com

***Abstract* - This paper presents some studies regarding the functional model of a robot system that has a visual servoing control, and also the simulation results for the system model. The information given by a visual system of the position of a target, materialized by a moving light source is used to realize the robot control.**

1. INTRODUCTION

The functional model of a robot system which presents a visual servoing control is presented in this paper. In fact, the robot system control is a position based visual servoing, the system using for control loop two CCD cameras. The robot block scheme is presented in figure 1.

As the block scheme of the robot system has a global regulator, which compares the real position with the prescribed one, in fact it is possible to say that the robot control is a global control. The robot system cinematic scheme for the system realized by authors is presented in figure 2. The robot system contains a Cartesian robot (TTT structures) respectively a visual subsystem materialized through two CCD cameras. The visual subsystem has to follows the robot end-effector (point G in figure 2). The CCD cameras present two DOF (pan and tilt).

The TTT robot joints are actuated by DC actuators, and the visual device are actuated by three step by step actuators (one generates the motion through Ox axes, and the other two generate the motions trough Oy_v and Oy_v' axes).

II. THE AIM OF THE FUNCTIONAL MODEL

The aim of the modeling and simulation of the robot system is to foresee how the system works before the effective realization of the system.

While the robot system is a complex system, which has to be dynamically controlled, and because its components are heterogeneous, some of them having continuous parameters while others discrete parameters, it was necessary to create the model for each conceived system component.

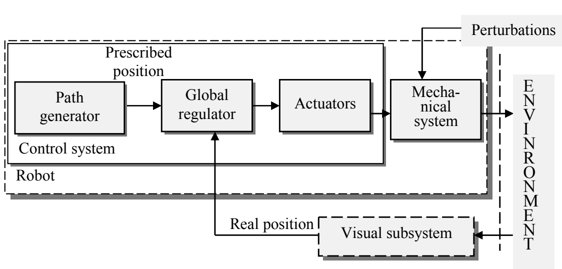


Fig. 1 The block scheme for a robot system which presents a position based visual servoing

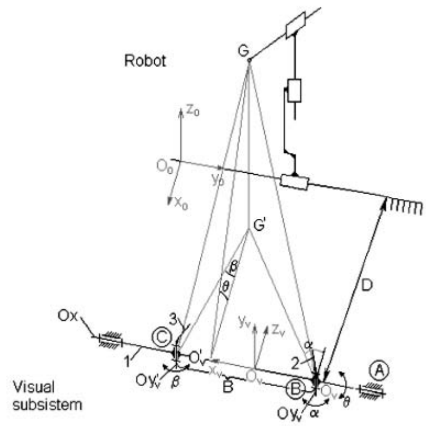


Fig. 2. The robot system kinematics' scheme

The authors generate the model for each component of the robot system to be able to establish:

- the acquisition time of the visual subsystem;
- the command execution time;
- the response time of the computer system,

all of those to establish if the robot system is working or not in real time.

In order to realize the model of the robot system behavior, a program which have the capability of the dynamic behavior redeem of the robot subsystems is needed. MATLAB® SYMULINK® possesses all the attributes necessary to model and simulate systems with time-modifying states.

III. THE SYSTEM COMPONENTS FUNCTIONAL MODEL

III.1. The DC motor functional model

The DC motor mathematical model can be found in [1]. The DC motor functional model contains two transfer functions "Electric", respectively "Load", which represents the Laplace transformations of the DC motors differential equations.

The "Load" block output is the motor angular velocity, which after the integration became the angular coordinate of that joint. Supply voltage represents the "Electric" block input.

Frictions during the motor function are modeled using the "Dry and viscous friction" block.

The aim of the block is the study of the dependency of its inputs with speed.

The DC motor model block scheme is presented in figure 3.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 19–24.

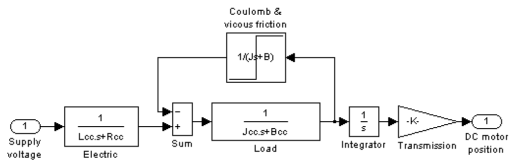


Fig. 3. The DC motor model block scheme

The main components of the DC motor model block scheme are:

- "Electric" represents the transfer function of the DC motor electric. The block input is the supply voltage [V] while the output is the torque [Nm].
- "Load" represents the transfer function of the DC motor mechanical component loaded with a resistant torque. The block input is the torque [Nm], while the output is the angular speed [rad/s].
- "Dry and viscous friction" block deals with viscous friction which depends on speed and dry friction. The input of the block is the angular speed and the output is the friction torque which is to be substituted from the torque.
- "Integration" is the block that realizes the integration of the angular speed necessary to obtain the generalized parameter of the corresponding cinematic couple. [rad].
- "Sum" is the block for two inputs addition.
- "Supply voltage" is the input for the D.C motor [V].
- "Position" represents the angular position of the motor shaft [rad].
- "Real speed" the angular speed of the motor shaft [rad/s].

III.2. The Step Motor Functional Model

Angular displacements of the visual sensor are made using step motors. Mathematic model of the step motor is presented in [3].

The step motors used in experimental device are 4 phase motors. The model for one winding is presented in figure 4.

- The following components were used:
- "In1" – command signal of the winding (step voltage impulses)
- "Winding 1 parameters" – the transfer function of the winding
- "Kmpp" constant of the motor
- "Kgpp" mechanical constant of the transmission
- "Winding 1 position" – function which models the winding position. Its expression is sin(n*u-v) where u is the input, which means the instant position of the rotor and v is the winding position.

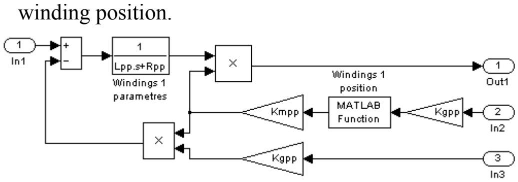


Fig. 4 The simulation model for one step motor winding

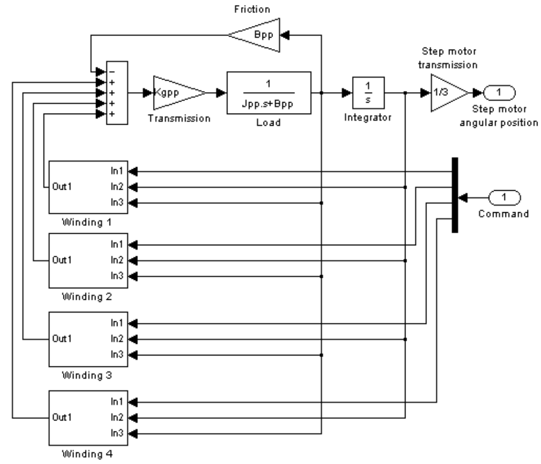


Fig. 5 The complete step motor model

- "±" – summing block
- "x" – multiplying block
- "Out1" – motor torque [daNm]
- "In2" – instant position of the rotor [rad]
- "In3" – instant speed of the rotor [rad/s]

The previously presented model is multiplied by 4 as can be seen in the complete motor model presented in figure 5.

Supplementary, the following blocks were added:

- "Load" – the transfer function which models the motor shaft loading behavior. It contains the inertia moment J, and the dry friction coefficient B. Their calculus is presented in [4].
- "Wear" – the transfer function which models the viscous friction
- "Integrator" – integration block having the same role as the one presented at the d.c. motor model.
- "Angular position MPP" – output of the step motor [rad].
- "Command signal" – port which transmits the command impulse to the windings phase-shifted for each winding.
- "Winding 'n'" – where n represents the winding number is the subsystem obtained from the model presented in fig. 5.
- "Transmission" – represents the mechanical module of the step motor
- "Step motor transmission" – represents a reduction gear with a single pair of tooth gear, having the gear ratio of 1/3.

III.3 Functional model of the visual sensor

Two major aspects were taken into account while realizing the visual sensor model: perspective transformation and discreet character of the image acquisition.

The following characteristic values were defined for the perspective transformation:

- the distance from the sensor lens focus point to the Ox_0y_o plane of the end-effector: "D" (in functional model "dist")
- angle between the instant position of the optical axe of the visual sensor and the perpendicular from the lens focus to the robot axle "α" (in functional model named "alfa");

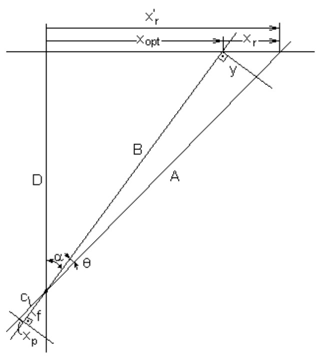


Fig. 6. Characteristic point position scheme and its correlation with visual sensor acquired image

- distance from the lens focus to the CCD plane, "f" ("dist_focal" in functional model);
- instant position " x'_r " of the characteristic point relative to an absolute measuring mark
- instant position of the sensor optical axes with the characteristic point plane intersection "x_{opt}"
- instant position of the characteristic point relative to the intersection of the optical axes of the sensor with the characteristic point plane x_r (named "u" in (6))
- the characteristic point instant position relative to the optical axes in image plane x_p.

Having these notations and using figure 6, following equations can be written:

$$B = \frac{D}{\cos\alpha} \tag{1}$$

$$x_{opt} = D \cdot tg\alpha \tag{2}$$

$$x'_r = D \cdot tg(\alpha + \theta) \tag{3}$$

$$x_r = x'_r - x_{opt} = D \cdot \left[tg\left(\alpha + arctg\frac{x_p}{f}\right) - tg\alpha \right] \tag{4}$$

$$x_p = f \cdot tg\left(arctg\left(\frac{x_r}{D} + tg\alpha\right) - \alpha \right) \tag{5}$$

Let's consider the initial moment 1 in which the target image in the image plane is located in its center and the visualizing sensor direction intersects the moving target plane in a point having the ordinate x_1.

Moving the target toward left, the corresponding displacement in the target image is toward right. As long as the target image is inside the interest rectangle, according to (2) and (3) instant position of the characteristic point relative to the intersection of the optical axes of the sensor with characteristic point plane can be calculated (image 2 figure 7).

If target image passes the interest rectangle perimeter a command is transmitted to the step motor in order to reposition the target image inside the interest rectangle (case 3).

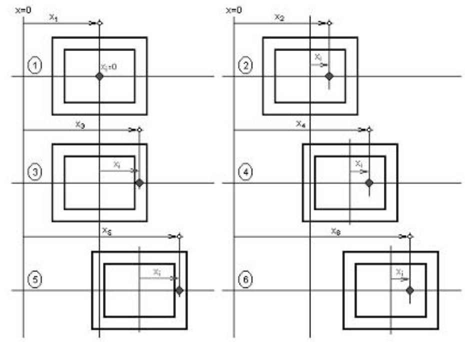


Fig. 7. Characteristic point position scheme and its correlation with visual sensor acquired image

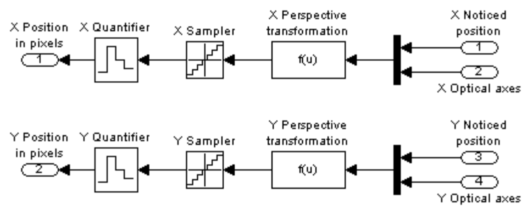


Fig. 8. Sensor functional model block scheme

The algorithm repeats, each time calculating the instant position of the characteristic point inside the interest rectangle using (2) and (3).

Functional model of the visual sensor is shown in figure 8. Component blocks are:

- "X or Y Noticed position" – input port which transmits the characteristic point position evolution
- "X or Y Optical axes position" – input port which transmits the optical axes with robot axles moving direction intersection position evolution
- "X or Y Perspective transformation" – Matlab® function that uses previously defined equations, having in simulation language the expression:

u=tan(atan(u/dist+tan(alfa))-alfa)*dist_focal*n_pixeli (6)

in which ,"u" represents the input of the ($u=x_r$) block

- "X or Y Sampler" – block used to model the discreet characteristic of the image acquisition board.
- "X or Y Quantifier" – is a quantifier of the x_r value defined at the beginning of the paragraph, which is a real value expressed as an integer number that represents the number of pixels that corresponds to x_r.
- "X or Y Position in pixels" – is the output port of the subsystem which transfers x_p to the command subsystem.

III.4. Functional modeling of the computation system

The computation system generates the command values for the robot actuators and for the positioning motors of the tracking visual system. Figure 9 presents the computation system model.

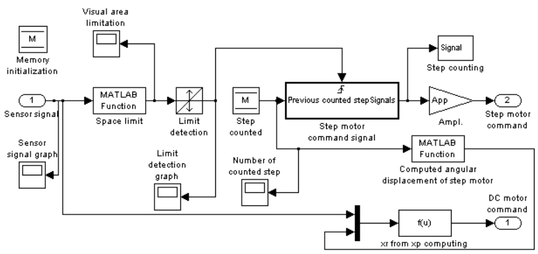


Fig. 9. Computation system model block scheme - subsystem X axes

In this figure "Sensor signal", "Limit reach signal" can be noticed; their role is to display the signals they receive in order to track the simulation evolution.

The mentioned command values are computed using the visual sensor transmitted value x_p as input.

The command signal of the step motor is generated when the limits of the image plane interest rectangle are overstepped, as shown in figure 7.

Inside the model this behavior is realized through the following blocks:

- "Memory initialization" – defines a space of memory in which system writes and reads the number of steps realized by the step motor.
- "Sensor signal" – system input port.
- "Space limit" – contains the relation:

$$u(1)\text{-}x\_frame*(u(2)+1) \quad (7)$$

where: - x_frame is the border of the limitation rectangle;
- u(1) is the input value of the visual sensor block
- u(2) is the number of steps executed by the step motor

- "Limit detection" – the block that initiates the function of the "Step motor command signal generator" at the moment of the border overstep.
- "Steps counted" – reads from memory the number of steps realized by the step motor
- "Step motor command signal" – subsystem having the structure shown in figure 10.

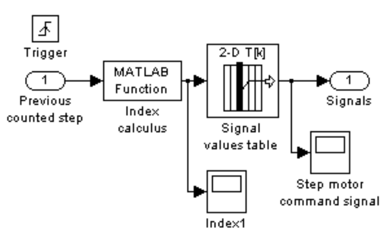


Fig. 10. Step motor command signal subsystem

TABLE 1
SIGNAL VALUE TABLE

Index	0	1	2	3	4	5	6	7
Winding 1	-1	0	0	0	1	0	0	0
Winding 2	0	-1	0	0	0	1	0	0
Winding 3	0	0	-1	0	0	0	1	0
Winding 4	0	0	0	-1	0	0	0	1

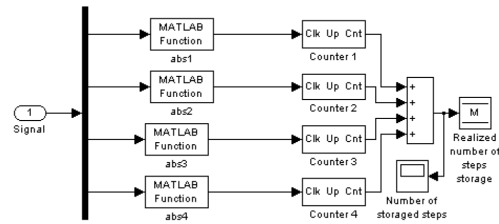


Fig. 11. Step counting block diagram

Input signal increments by 1 after which the function:

$$\mathrm{Mod}(u+1,8) \quad (8)$$

is applied.

Function (8) is the rest of the dividing with 8 of the incremented signal, such computing the index of the column of the table memorized in the "Signal values table" block. Values memorized in this block are shown in table 1.

Depending on the index value (position of the winding which is to be actuated) four signal are generated, one for each winding. These signals are transmitted through the "Signals" port to the computing system.

- "Computed angular displacement of the step motor" is a block which calculates the "alfa" parameter with:

$$alfa=step\_number*angular\_step \quad (9)$$

where *step_number* is the number of steps realized by the step motor and *angular_step* is the value of a single step of the step motor [rad].

- "Alfa" parameter is needed to compute the level of the DC motor command value.
- "Step counting" is a subsystem which counts the steps realized by the step motor and transfers the information in memory.

The block diagram of the subsystem is presented in figure 11. The meanings of the figure 11 notations are:

- "Signal" – input system port;
- "abs1",…"abs4" – returns the absolute value of the signals on the 4 channels, corresponding to the 4 windings;
- "counter1",…,"counter4" – counters having as input "Clock" (clk) and as output "Count" (cnt) block;
- "Sum" – computes the sum.
- "Realized number of steps storage" writes in memory the computed value.
- "Step motor command" – output port from the command system which transmits the generated signals toward step motor
- "D.C. motor command" – is the output port of the command system which transmits the "x_r" parameter value to the DC motor actuating devices of the robot.
- "x_r from x_p computing" – the d.c. motor signal command component is computed with (4) which in modeling language can be written:

$$dist*(tan(alfa+atan(u/dist\_focal))-tan(alfa)) \quad (10)$$

III.5 Functional model

In the previous paragraphs functional models of the component elements of the robot system were presented. These elements integrate in the complete schema of the model.

The functional model of the robot was realized and is presented in figure 12.

"Commanded position" block generates a step signal which represents the programmed position of the axles.

At the receiving time of the signal "Chopper" block generates the signal that represents the necessary supply voltage for the DC motor. "X or Y DC motor" block has as output the signal that represents the relative position of the modeled cinematic couples elements.

This position is seized by the "visual sensor" and evaluated by the "computing system", being after compared with the programmed value.

The comparison result is the position error and represents the new command level. As the value that characterizes the realized position increases, the difference between the programmed values and the real value decreases, and as a fact the command voltage of the d.c. motor decreases, so that in the end the d.c. motor stops in the programmed position.

Because the visual field of the sensor doesn't fit the entire moving domain of the modeled axle it moves in such a way that the characteristic point always remains in the visual field of the sensor.

The displacement is realized using a step motor commanded by the computing system in such a manner that it executes a step each time the characteristic point approaches to a certain distance from the sensor visual field limit.

The structure and the functioning mode of certain modules have been described previously.

To evaluate the behavior of the modeled system in the block schema, graphical displaying elements were introduced for different signals, as functions of time ("Real position", "Optic axes position", et).

Robot's actuators are the two DC motors ("X DC Motor", "Y DC Motor") having the feedback assured by the same visual sensor ("Visual sensor") and computation system ("computation system").

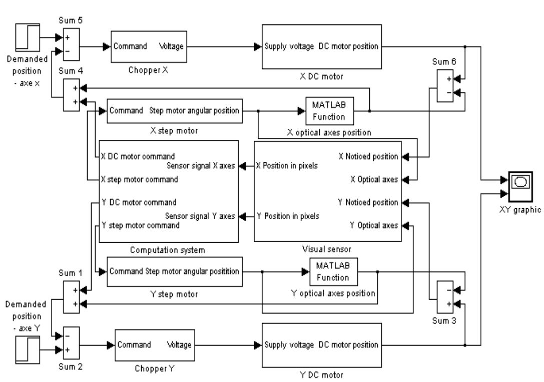


Fig. 12. Functional model of the robot sensorial driving system

Each of the actuators generates the relative motions of their corresponding cinematic couple, each of them having programming elements ("Demanded position – axe X", "Demanded position – axe Y"), execution elements ("Chopper X", Chopper Y"). The step motors ("X step motor", "Y step motor") assure the visual sensor compensation movements, while the sensors' optical axes are computed by the computing blocks ("X–optical axes position", "Y–optical axes position"). The sum blocks "Sum2" and "Sum5" assures the feedback, while "Sum1", "Sum3", "Sum4" and "Sum6" realize the compensations appeared due to the visual targeting system follow of the robot's characteristic point.

The real path of the robot's characteristic point can be viewed using the displaying block "XY graph". This configuration can realize only linear trajectories because the signal command generators for X and Y axes generates constant signal.

IV. Simulation Results on the Experimental Model

The results of the robot system simulation for two cinematic couples that have relative movements along O_0x_0 and O_0y_0 axes were achieved for a step signal having the shape presented in figure 13.

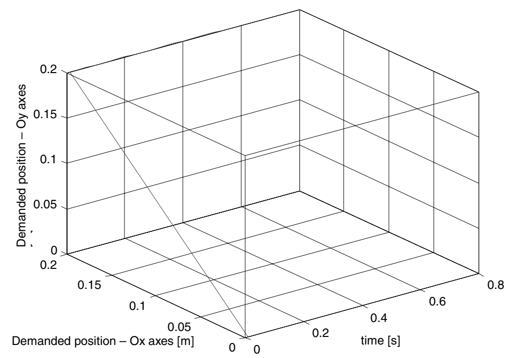


Fig.13. The step command signal

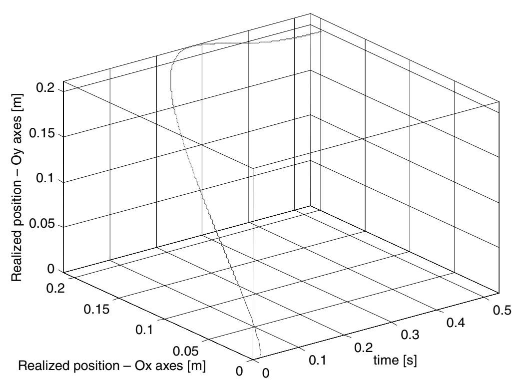


Fig.14. Robot system answer to the step command signal

This signal represents the prescribed position of the lightning target along O_0x_0 and O_0y_0 axes. The command signal was generated with a delay of 0.01 [s]. The signal is applied to the chopper which will generate a command voltage.

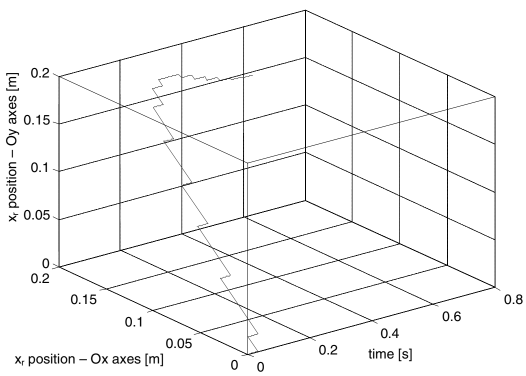


Fig. 15. Instant x_r position variation versus time

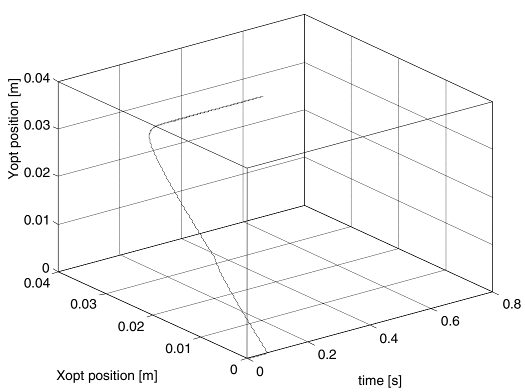


Fig.16. x_{opt} signal variation versus time

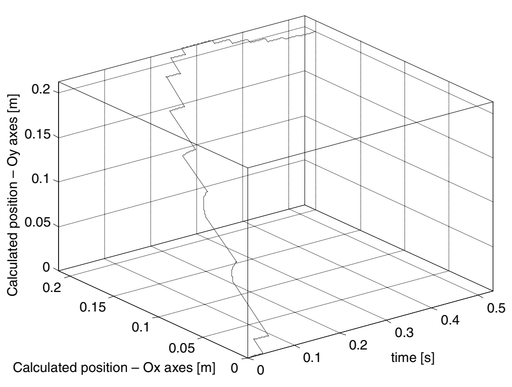


Fig. 17. Computed position variation with time

Figure 14 presents the robot system answer to the previously presented step command signal. This signal represents the time variation of the targeted position and it is collected at the output port from the DC motor blocks.

It was noticed that the setting-up time after which the robot system reaches the prescribed position of 0.2 [m] with an error of 2.5% is 0.4 [s]. If the prescribed positioning error needs to be under 1%, setting-up time increases to 0.5 [s].

During target displacement "Computing system" block concluded the necessity of two command signals for the step motor so that the target image should be placed inside the interest rectangle from the image plane.

The step motor output signal is transformed in "Sensor optical axes position modification" block into a signal corresponding to x_{opt} computed with (2), having the evolution with time presented in figure 16.

Over this signal the signal presented in figure 15 is superimposed, thus the position detected by the mean of the visual sensor (real position) being obtained. Figure 17 presents this signal.

V. CONCLUSIONS

As can be seen from the robot system response to a step signal input the time needed to achieve the prescribed position with 1% error is 0.53 [s], time in which the adjustment loop of the system was run for 16 times, so, the average time for one transition of the loop is 0.033 [s].

Table 2 presents the times achieved during the robot system functioning simulation.

Concluding, considering the times obtained during the simulation process, it can be said that the robot system, which uses information from a visual tracking system, functions in real time.

TABLE 2
TIMES MEASURED DURING SIMULATION

Simulated process	Times obtained during simulation
DC motor signal generating	0.01 [s]
Image acquisition	0.03 [s]
Compute in „Computing system" block	0.01 [s]
Step motor signal generating	0.01 [s]

REFERENCES

[1] Bogdanov, I., (1989) Microprocesorul în comanda acționărilor electrice, Editura Facla, Timişoara.
[2] Hashimoto, K.,: *Visual servoing: Real Time Control of Robot Manipulators Based on Visual Sensory Feedback*, World Scientific Series In Robotics and Automated Systems - vol. 7, Singapore.
[3] B. J. Nelson, A Distributed Framework for Visually Servoed Manipulation Using an Active Camera, International Journal of Robotics and Automation, 2002.
[4] Țarcă, R.,: *Visual Closed-Loop Robot Control*, PhD Thesis, Polytechnic University of Timişoara, 2001.
[5] Corke, P.I. (1994) High-Performance Visual Closed-Loop Robot Control, PhD Thesis, University of Melbourne.
[6] Hashimoto, K. (1991) Model-Based Control of Manipulator with Image-Based Dynamic Visual Servo, PhD Thesis, Osaka University.

An Approach to the Design of Optimal Test Scheduling for System-On-Chip Based on Genetic Algorithm

P. SAKTHIVEL
Ramanujan Computing Centre
Anna University
Chennai – 600 025
INDIA
E-Mail: psv@annauniv.edu

P. NARAYANASAMY
Department of Computer Science and Engineering
Anna University
Chennai – 600 025
INDIA
E-Mail: sam@annauniv.edu

Abstract- **Test Methodologies for Globally Asynchronous Locally Synchronous (GALS) System On a Chip (SOC) are a subject of growing research interest since they appear to offer benefits in low power applications and promise greater design modularity. Pre-designed cores and reusable modules are popularly used in the design of large and complex Systems. As the size and complexity of System increase, the test effort, including test development effort, test data volume, and test application time, has also significantly increased. Available techniques for testing of core-based systems on a chip do not provide a systematic means for compact test solutions. A test solution for a complex system requires good optimization of Test Scheduling and Test Access Mechanism (TAM). In this paper, we provide a Test Scheduling Optimization for Globally Asynchronous Locally Synchronous System-On-Chip Using Genetic Algorithm that gives compact test scheduling.**

I. INTRODUCTION

Very Large Scale Integrated (VLSI) circuits designed using modern Computer Aided Design (CAD) tools are becoming faster and larger, incorporating millions of smaller transistors on a chip [1]. VLSI designs can be divided into two major classes: Synchronous and Asynchronous circuits. Synchronous circuits use global clock signals that are distributed throughout their sub-circuits to ensure correct timing and to synchronize their data processing mechanisms [14, 15]. Asynchronous circuits contain no global clocks. Their operation is controlled by locally generated signals [13]. Asynchronous circuits [11] have many potential advantages over their synchronous equivalents including lower latency, low power consumption, and lower electromagnetic interference [11].

However, their acceptance into industry has been slow, which may be due to a number of reasons. System On a Chip technology is the packaging of all the necessary electronic circuits and parts for a "System" on a single integrated circuit generally known as a "Microchip"[5]. System On a Chip technology is used in small, increasingly complex consumer electronic devices. Some such devices have more processing power and memory than a typical computer. The design of asynchronous circuits has been attracting more interest recently, as clock distribution on a large die becomes increasingly difficult. The ITRS road-map [11] predicts that, as a solution to the clock distribution problem, Globally Asynchronous Locally Synchronous system will become mainstream in the near future. In a GALS system, a number of synchronous islands of logic communicate asynchronously using a suitable interconnect. Unfortunately, the testability of asynchronous systems is considered one of their major drawbacks.

Testing SOC becomes an increasing challenge [5] as these devices become more complex. An SOC design is typically built block by block. Efficient testing is also best done by block by block. Recently, pre-designed cores are also used in the SOCs [6]. Testing individual circuits, individual blocks and individual cores have established technologies. But, available techniques for testing of core-based systems on a chip do not provide a systematic means for synthesizing low overhead test architectures and compact test solutions [7]. Embedded cores such as processors, custom application specific integrated circuits, and memories are being used increasingly to provide SOC solutions to complex integrated circuit design problems [8]. The advance in design methodologies and semiconductor process technologies has led to the development of systems with excessive functionality implemented on a single chip [7].

In a core based design approach, a set of cores, that is predefined and pre-verified design modules, is integrated into a system using user defined logic and interconnections. In this way, complex systems can be efficiently developed [9]. However, the complexity in the system leads to high-test data volumes and design and optimization of test solution are must for any test. Therefore consider the following independent problems [7]:

- How to design an infrastructure for the transportation of test data in the system, a test access mechanism.

- How to design a test schedule to minimize test

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 25–28.

time, considering test conflicts and power constraints.

The testable units in an SOC design are the cores, the User Defined Logic (UDL) and the interconnections. The cores are usually delivered with predefined test methods and test sets, while the test sets for UDL and interconnections are to be generated prior to test scheduling and TAM Design. The workflow when developing an SOC test solution can mainly be divided into two consecutive parts: an early design space exploration followed by an extensive optimization for the final solution.

In this paper, we propose a new technique using Genetic Algorithm for optimizing the test scheduling for Globally Asynchronous Locally Synchronous System On Chip with the objective to minimize the test application time. The aim with our approach is to reduce the gap between the design space exploration and the extensive optimization that is to produce a high quality solution in respect to test time and test access mechanism at a relatively low computational cost.

The rest of the paper is organized as follows. Various issues related to SOC testing and Test Scheduling Techniques are discussed in Section 2. Genetic Algorithm based framework for Test scheduling is presented in Section 3. In section 4, the experimental results for ITC-02 Benchmark SOC circuit p34392 are presented. Finally, section 5 gives conclusion to the paper.

II. SOC TESTING AND TEST SCHEDULING TECHNIQUES

The test-application time when testing a system can be minimized by scheduling the execution of the test sets as concurrently as possible [1]. The basic idea in test scheduling is to determine when each test set should be executed, and the main objective is to minimize the test application time.

The scheduling techniques can be classified by the following scheme [2]:

- No partitioned testing
- Partitioned testing with run to completion, and
- Partitioned testing.

A. Testing System On a Chip

Integration of a complex system, that until recently consisted of multiple Integrated Circuits, onto a single Integrated Circuits, is known as System On a Chip [3]. The shrinking of silicon technology leads to increase in number of transistors on a chip. This increases the number of faults and test vectors that in turn leads to the serious increase in test time. Test time reduction is one of the research challenge [4] in the SOC design paradigm.

The most important issues in the System On a Chip Testing are as follows [5]:

- Controlling the whole process of SOC Testing.
- Testing the User Defined Logic and Interconnections.
- Testing cores with different functionalities coming from different vendors.
- Accessing cores from the system's primary inputs and primary outputs.

B. Test Access Mechanism

The test access mechanism (TAM) takes care of chip test pattern transport. It can be used to transport test stimuli from the test pattern source to the core under test and to transport test responses from the core under test to the test pattern sink. The TAM is by definition implemented on chip [6].

The following are the four problems structured in order of increasing complexity [2].

- P_W: Design a wrapper for a given core, such that the core testing time is minimized, and the TAM width required for the core is minimized.
- P_{AW}: Determine (i) an assignment of cores to TAMs of given widths, and (ii) a wrapper design for each core such that SOC testing time is minimized.
- P_{PAW}: Determine (i) a partition of the total TAM width among the given number of TAMs, (ii) an assignment of cores to TAMs of given widths, and (iii) a wrapper design for each core such that SOC testing time is minimized.
- P_{NPAW}: Detremine (i) the number of TAMs for the SOC, (ii) a partition of the total TAM width among the given number of TAMS, (iii) an assignment of cores to TAMs of given widths, and (iv) a wrapper design for each core such that SOC testing time is minimized.

The above problems are all NP – Hard problems. Therefore, efficient heuristics and other techniques are needed for large problem instances [7]. In this work, we are presenting Genetic Algorithm based approach [8] to effectively solve the problems namely P_{AW} and P_{PAW}.

III. GENETIC ALGORITHM BASED FRAMEWORK FOR TEST SCHEDULING

In this section the Genetic Algorithm (GA) that is used for generating test sequences for System On a Chip is described [1,2,3,4,9]. First, the basic idea of the method is given. Then we present the representation of test conditions and the objective function and provide some insight into the parameter settings of the Genetic Algorithm [10]. Genetic algorithms can be used to solve effectively the search and optimization problems. GAs consists of population of solutions called chromosomes. Here the chromosomes are an encoding of the solution to a given problem. The algorithm proceeds in steps called generations. During each generation, a new population of individuals is created from the old, by applying genetic operators. Given old generation, new generation is built from it, according to the following operation given in section 3.1, 3.2 and 3.3 [11].

A. Selection

This operator selects the individuals from the old generation. The individual with a better performance possess higher chance of getting selected.

B. Crossover

This operator generates two new chromosomes from the couple of selected chromosomes. A random point on the chromosome also known as cross-site is selected. Portions of individuals in a couple are exchanged between themselves to form new chromosomes as follows [12]:

for I = 1 to number of entries in the chromosome

Child (I) = Parent1 (I) if I <= cross-site
= Parent2 (I) if I > cross-site

Child2 (I) = Parent2 (I) if I <= cross-site
= Parent1 (I) if I > cross-site

C. Mutation

This operator chooses a random chromosome and modifies it to form the new chromosome.

D. Overview of our Method

The different steps of our method is given as follows [1]:

1. Generate the initial population of chromosomes, randomly.
2. Sort the initial population in ascending order of the cost.
3. While there is no improvement in cost function
   Do
   Select first 20% chromosome as best class.
   Generate 40% chromosomes using crossover operator.
   Generate 40% chromosomes using mutation operator
   Sort this generation in ascending order of the cost.
4. End of genetic algorithm.

IV. EXPERIMENTAL RESULTS

Our experiments were conducted for the ITC-02 SOC benchmark circuit p34392 [3]. The p34392 SOC benchmark circuit consists of 20 modules, of which 15 are combinational and 5 are sequential circuits. It has 3 levels, 2057 input and outputs, 20948 scan flip-flops and 66349 test patterns.

The Table - 1 gives the result for SOC p34392 with two partitions. W is the width of TAM. w1 and w2 are size of the partition 1 and partition 2. The processor time and test time given under the Heuristics are taken from [2]. The processor time and test time given under Genetic Algorithm is the combination of the results of our experiment and [1]. For two partitions of total TAM width, the maximum processor time taken is 1.34 seconds and minimum processor time taken is 0.82 seconds.

TABLE – 1: RESULTS FOR SOC P34392 WITH TWO PARTITIONS

W	w1 + w2	(Processor Time(*Seconds*))/ Test Time(*Cycles*)	
		Heuristics	Genetic Algorithm
16	8 + 8	(1)/ 1080940	(1.34)/ 1080900
24	15 + 9	(1)/ 928782	(1.25)/ 928562
32	21 + 11	(1)/ 750490	(1.10)/ 749850
40	24 + 16	(1)/ 721566	(0.97)/ 721450
48	31 + 17	(1)/ 709262	(0.85)/ 708550
56	38 + 18	(1)/ 704659	(0.82)/ 704650
64	18 + 46	(1)/ 700939	(0.88)/ 700800

V. CONCLUSION

The experimental results are given for only one Benchmark circuit SOC p34392 with two partitions. The result gives good approximation compare to Heuristics within a few generations with acceptable processor times. This establishes the suitability of this problem to be solved by Genetic Algorithm. We can apply this technique to all the other SOCs given in [3] having more number of cores with many scan chains and even more number of TAM widths.

REFERENCES

[1] S. Chattopadhyay and K. Sudharsana Reddy, "Genetic Algorithm Based Test Scheduling and Test Access Mechanism Design for System-On-Chips", *Proceedings of the International Conference on VLSI Design*, 2003.

[2] Vikram Iyengar, K. Chakrabarthy and Erik. J. Marinissen, "Efficient Wrapper/TAM Co – Optimization for Large SOCs", *http://www.ee.duke.edu/~krish/237_iyengar_v.pdf*

[3] Erik Jan Marinissen, V. Iyengar and K.Chakrabarthy,"A Set of Benchmarks for Modular Testing of SOCs", *http://www.extra.research.philips.com/itc02socbenchm*

[4] Martin Keim, Nicole Drechsler, Rolf Drechsler and Brend Becker, "Combining GAs and Sysmbolic Methods for High Quality Tests of Sequential Circuits", *Journal of Electronic Testing: Theory and Applications*, Vol. 17, 2001, pp 37-51.

[5] Srivaths Ravi, Ganesh Lakshminarayana and Niraj. K. Jha, "Testing of Core-Based Systems-on-a-Chip", *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, Vol. 20, No. 3, 2001, pp. 426-439.

[6] Erik Larsson, Klas Arvidsson, Hideo Fujiwara and Zebo Peng, "Efficient Test Solutions for Core-Based Designs", *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, Vo. 23, No. 5, 2004, pp. 758-774.

[7] Anuja Sehgal, V. Iyengar and K. Chakrabarthy, "SOC Test Planning Using Virtual Test Access Architectures", *IEEE Transactions on Very Large Scale Integration Systems*, Vo. 12, No. 12, 2004, pp. 1263 – 1276.

[8] Yuejian Wu and Paul MacDonald, "Testing ASIC with Multiple Identical Cores", *IEEE Transactions on Computer Aided Design of Integarated Circuits and Systems*, Vo. 22, No. 3, 2003, pp. 327 – 336.

[9] V. Iyengar, A. Chandra, S. Schweizer and K. Chakrabarthy, "A Unified Approach for SOC Testing Using Test Data Compression and TAM

Optimization", *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, 2003.
[10] Cheng-Wen Wu, "SOC Testing Methodology and Practice", *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, 2005.
[11] Aristides Efthymiou, John Bainbridge and Douglas Edwards, "Adding Testability to an Asynchronous Interconnect for GALS SOC", *Proceedings of the 13th Asian Test Symposium*, 2004.
[12] Hemangee K. Kapoor and Mark B. Josephs,"Modelling and Verification of delay-insensitive circuits using CCS and the Concurrency Workbench", *Information Processing Letters*, Vo. 89, 2004, pp.293-296.
[13] Eunjung OH, Soo-Hyun KIM, Dong-Ik LEE and Ho-Yong CHOI, "High Level Test Generation for Asynchronous Circuits from Signal Transition Graph", *IEICE Transactions on Fundamentals*, Vo.E85-A, No. 12, 2002, pp. 2674 – 2683.
[14] Yin-He SU, Ching-Hwa CHENG and Shih-Chieh CHANG, "Novel Techniques for Improving Testability Analysis", *IEICE Transactions on Fundamentals*, Vol. E85-A, No. 12, 2002, pp. 2901-2912.
[15] Mathew Sacker, Andrew D. Brown, Andrew J. Rushton and Peter R. Wilson, "A Behavioral Synthesis System for Asynchronous Circuits", *IEEE Transactions on Very Large Scale Integration Systems*, Vol.12, No.9, 2004, pp. 978 – 994.

P. Sakthivel received the Bachelor of Engineering Degree in Computer Science and Engineering from Sathyabama Engineering College, University of Madras, Chennai, India in 1992 and Master of Computer Science and Engineering Degree from Jadavpur University, Kolkata, India in 1994. He is currently Lecturer in Ramanujan Computing Centre, Anna University, Chennai, India and working towards the PhD Degree in the Department of Computer Science and Engineering, Anna University, Chennai, India. His research interests include VLSI Design and Testing and Computer Aided Design of VLSI Circuits.

P. Narayanasamy received the Bachelor of Engineering Degree in Electrical and Electronics Engineering from Coimbatore Institute of Technology, University of Madras, Coimbatore, India in 1980, Master of Engineering Degree in Electrical and Electronics Engineering from Anna University, Chennai, India in 1982 and PhD Degree in the area of Computer Applications in Electrical Engineering from the Faculty of Electrical Engineering, Anna University, India in 1989. He is currently Professor and Head of the Department of Computer Science and Engineering, Anna University, Chennai, India. His research interests include VLSI Design and Testing, Computer Communication Networks, Wireless and Mobile Computing. He has guided many students for PhD and MS Programmes. He has published many technical and research papers in the National and International Conferences and Journals.

Vision-based Monitoring System for Detecting Red signal crossing

Ritesh Sharma & Dr. Sreela Sasi Ph.D.
Gannon University
Erie, PA 16541, USA
{sharma002, sasi001}@gannon.edu

Abstract

Red light running is a leading cause of urban crashes that often results in injury and death. The main reason for this is when the automobile driver fails to stop at the intersections when the signal light is red, and runs over other road users. In this research a computer vision-based low cost system is proposed for continuously monitoring the vehicles crossing the intersections when the signal light is “red” with the help of video cameras placed at the intersections, and penalizing careless drivers accordingly. This monitoring system finds application at all busy intersections. By using this setup people would be more conscious of getting penalty tickets which in turn will discourage them from running red lights. This research is intended to provide a support tool for the law enforcement agencies to proactively ensure that intersections are engineered to discourage red light running.

Keywords: Vision-based Monitoring System, red signal light crossing, vehicle identification number.

1. Introduction

Red light running is a leading cause of urban crashes that often results in injury and death. Total road deaths in USA in year 2004 were 42,636. A survey conducted during 1999-2000 revealed that 20% of vehicles involved in road accidents did not obey the signal. Each year “red” light running causes nearly 200,000 accidents resulting in above 800 deaths and 180,000 injuries [1], [2]. Signal lights on the road intersection are for controlling traffic. Some people do not abide by the traffic rules and cross the intersection when the signal light is ‘red’. Figure 1 shows a negligent driver, who does not stop at the red signal and risks the lives of other drivers.

Figure 1: An accident scene on the red light signal intersection

To reduce the accident rate at the intersections, busy and accident prone intersections should be monitored. Not all of the intersections can be monitored 24x7 by the authorities. This demands a cost effective and automated system for continuously monitoring intersections and penalizing the people who violate the traffic rules.

Automatic License Plate Recognition (ALPR) systems have been developed and discussed in [3], [4], [5], [6]. In ALPR systems for monitoring intersections, there is a still camera for capturing the license plate of the car on the intersection. Sensors are located on the road to detect the presence of a vehicle at the intersection. When the signal light is ‘red’ and the sensors are active then a still photograph is taken which is used for issuing penalty ticket by the law enforcement authorities. The camera is accompanied with a bright flash which helps in image quality and cautions the driver for his/her

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 29–33.

violation [7]. The ALPR system is not a foolproof system because the license plate can be tampered or the plate might be stolen from another car or due to bad weather conditions the license plate could not be visible, or the sensors on the road might be tampered.

In this research an expert system that would capture the Vehicle Identification Number (VIN) and the License plate of the vehicle crossing the intersection on 'red' signal using two video cameras placed intersection and will send it to the authorities for action, is proposed. Using the vehicle identification number (VIN), it is possible to find the owner of the car, insurance details and the car facts report. The Vehicle Identification Number (VIN) is usually found on the left side of the dashboard under the windscreen, and is shown in Figure 2, enclosed in the red ellipse.

Figure 2: The vehicle identification number on the dash board below the front windscreen

No sensors are needed for the novel Vision-based Monitoring System for Detecting Red signal crossing (VMSDR) that captures and recognizes both Vehicle Identification Number (VIN) and License plate of the vehicle running 'red' signal light at the intersections. VMSDR system needs two video cameras out of which one is placed on the sidewalk and the other is placed on the pole above the intersection adjacent to the signal lights. The video camera placed on the side walk captures the license plate and the video camera placed on the pole along with the 'signal light' captures the VIN. The positions of the cameras at the intersection are demonstrated in Figure 3.

Figure 3: Position of the video cameras on the intersection

The video cameras will record only during the period when the signal light is 'red' using the timer information. These frames are continuously processed using a processing unit attached to the camera. The processing unit detects the Vehicle Identification Number (VIN) and the corresponding license plate number from the video frames obtained from both video cameras. This information is sent to the base station at the municipal corporation where using this information the owner of the vehicle, address, recent penalty tickets and insurance details are identified and penalty tickets can be issued. The owner will be given a period of time for informing the authority in case someone else is driving his car, during the time and date mentioned in the ticket.

2. VMSDR Architecture

A high level architecture for the VMSDR system is proposed, and is shown in Figure 4(a).

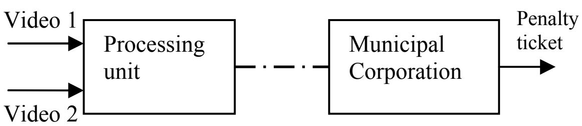


Figure 4(a): High Level VMSDR Architecture

A detailed diagram is shown in figure 4(b).

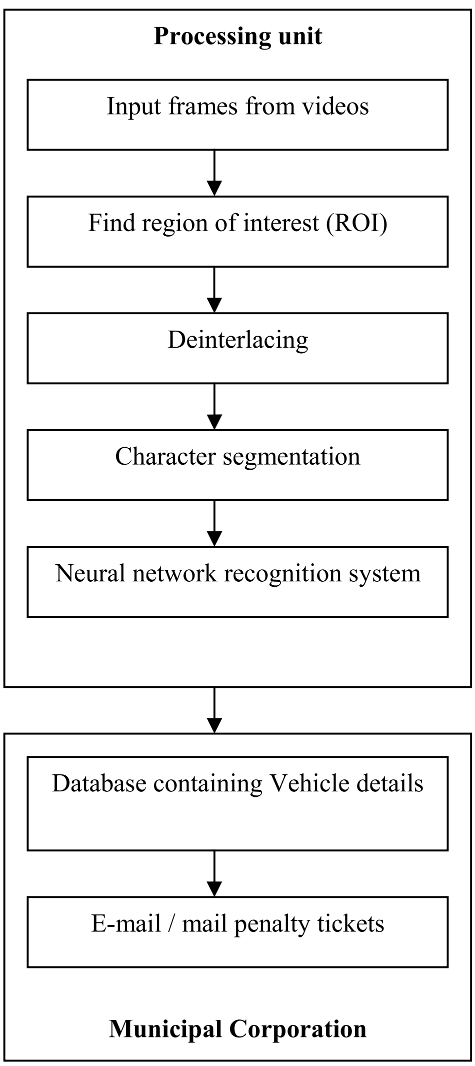


Figure 4(b): Detailed VMSDR architecture

3. Simulation

For simulating the VMSDR architecture a video camera is placed at a height of 10 feet above the ground on a fixed pole in the parking lot at Gannon University, and another camera on the sidewalk. The timer of the processing unit is used for synchronization of videos from both the video cameras. The video captures 25 frames per second. These are used to identify the vehicle identification number (VIN) and the license plate number. As an example a single frame is used to explain the steps for detecting the VIN.

Initially the region of interest (ROI) that includes the VIN is cropped from the video frame and is shown in the Figure 5.

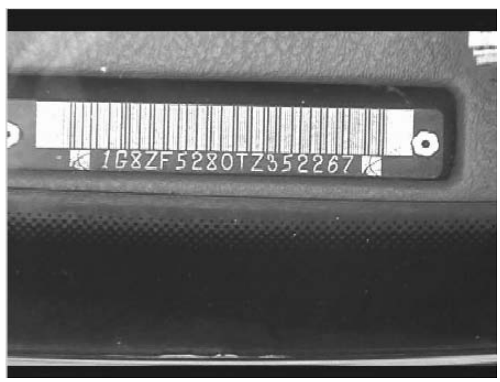


Figure 5: Region of interest that includes the VIN on the metallic plate

The ROI is further narrowed down to only the VIN and is used for further processing. A fine cropping is performed that includes only the VIN of size 24x135. Matlab 7.04 is used for simulating the algorithms. This code can be embedded in the processing unit attached to the video camera unit. The ROI consisting of only the VIN is shown in Figure 6 inside a matlab window.

Figure 6: Region of interest (ROI)

Since the vehicles may be moving fast it is likely to have interlacing artifacts in the ROI. Interlacing [8] can cause the image to have the artifacts such as Combing effect, Mice teeth, Saw tooth edge distortion, Interlaced lines, Ghost image, Blurring, etc. In order to remove these artifacts, a deinterlacing technique using linear interpolation is used. This will refine the characters in the VIN which are used for character recognition purpose.

The ROI obtained after deinterlacing is used for character recognition. The characters are segmented into a 7x5 matrix and are stored in arrays. Figure 7 shows some of these character images.

Figure 7: Character images sent to neural network

Each character array is fed to a neural network which recognizes the characters of the VIN. The neural network is trained [9] for 26 alphabets and 10 digits for both noise and without noise and uses backpropagation algorithm to recognize the array. Figure 8 shows the result of feeding the image consisting of “1” to the neural network.

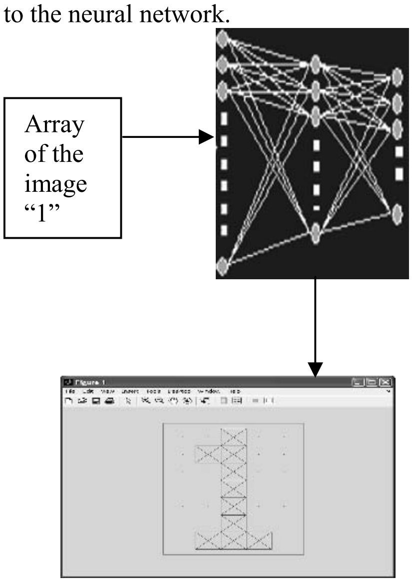


Figure 8: Character images recognized by neural network algorithm

All the images are fed to the neural network one after another and the complete VIN is recognized. The procedure is repeated for at least 10 frames for each vehicle.

The license plate is recognized using the algorithms given in [10], [11] and is sent along with the recognized VIN and the 10th frame from video camera as a proof of the vehicle in the middle of intersection on red light, to the test database management system which had data for 20 people.

The VIN and the license plate numbers are verified using this database containing the vehicle’s details such as VIN, License plate number, owners name and address, insurance details, tickets issued in the current year. Using the address of the owner, a penalty ticket is issued based on his/ her previous driving records. A time period is given to the owner to contest / challenge in case someone else was driving during the ticketed time. In this way the driver is penalized instead of the owner.

Instead of a test database management system used for simulation, a fully pledged Database Management System can be used at the municipal corporation side and penalty tickets can be issued automatically through e-mail or using ordinary mail. In case more details are needed, car facts of the vehicle can be checked using the VIN.

4. Conclusion

This system is very useful in identifying careless drivers who risk the life of innocent people and other drivers on the intersections, and penalize them round the clock.

The system if implemented would reduce the man power needed to guard the busy intersections and to reduce number of persons employed for issuing tickets. Drivers are forced to be careful against running ‘red’ signals because tickets are issued with proof of time and images of VIN and license plate of the vehicle in the middle of the intersection when the signal light is red.

The limitations of this system are the extreme weather conditions. If there is snow on the windscreen right above the VIN or if there is a sticker which obstructs the VIN as seen from the camera, the VIN cannot be extracted and recognized.

Future work will focus on having access to the VIN details of all the vehicles in 51 states which are distributed across the country. This

This page intentionally blank

system can monitor tickets issued in another state to the same driver. The VMSDR system can be extended to include the details of drivers who rent vehicles from rental services from in state or out of state

5. References

[1] http://www.driveandstayalive.com/info%20section/statistics/stats-usa.htm

[2] http://www.dft.gov.uk/stellent/groups/dft_control/documents/homepage/dft_home_page.hcsp

[3] http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

[4] http://www.motorola.com/governmentandenterprise/northamerica/en-us/solution.aspx?navigationpath=id_801i/id_826i/id_2694i/id_2695i

[5] http://www.cctv-information.co.uk/constant3/anpr.html

[6] http://www.oletc.org/oletctoday/0415_licplate.pdf#search=%22automatic%20license%20plate%20recognition%22

[7] http://www.intertraffic.com/marketplace/mypage/products_detail.asp?mypageid=1656&productid=2462

[8] http://en.wikipedia.org/wiki/Interlacing

[9] http://www.math.ufl.edu/help/matlab/ReferenceTOC.html

[10] J.R. Parker, Practical Computer Vision Using C, Wiley 1994, New York.

[11] R.A. Lotufo, A.D. Morgan and A.S. Johnson. Automatic Number Plate Recognition. IEE Colloquium on Image Analysis for Transport Applications, London, February 1990.

Describing Function and Error Estimation for Class of Nonlinear Systems with Fuzzy elements

Tomislav Sijak
Control Engineering Dept.
Brodarski Institute
Av.V.Holjevca 20,
10020 Zagreb, Croatia
tsijak@hrbi.hr

Ognjen Kuljaca
Dept. of Advanced Technology
Alcorn State University
1000 ASU Drive 360
Alcorn State, MS 39096-7500
okuljaca@alcorn.edu

Ljubomir Kuljaca
Dept. of Control and Computer
Engineering in Automation
Faculty of El.l Eng.and Computing
Unska 3, 10000 Zagreb, Croatia
ljubomir.kuljaca@fer.hr

Abstract **– The procedure for approximate analytical determination of describing function of nonlinear systems with odd static characteristics is presented in the paper. Generalized mathematical expressions for determining such describing function with error estimation are given. The procedure is illustrated on determination of describing function, and corresponding error estimation, of Mamdani fuzzy element and nonlinear system with fuzzy element and saturation nonlinearity.**

I. Introduction

Describing function is an equivalent gain of nonlinear element, defined by the harmonic linearization method of nonlinear static characteristic [4, 7 and others]. It is a known method of analysis and synthesis when nonlinear system can be decoupled into linear and nonlinear parts (Fig.1). If the linear part of the system has the characteristics of low-pass filter and if we apply periodical signal to the system, output signal will have the same base frequency as input signal with damped higher frequencies.

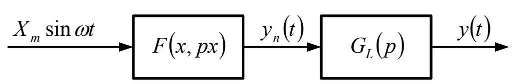


Fig.1 Nonlinear system represented with decoupled nonlinear $F(x, px)$ and linear parts $G_L(p)$, $p = d/dt$.

If the amplitudes of higher harmonics are relatively small to the amplitude of the first harmonic, output signal can be approximated by its first harmonic. Mathematically, first harmonic of the output signal, for the sinusoidal input signal $X_m \sin \omega t$, can be expressed by the Fourier expressions:

$$\begin{aligned} y_N(t) &\approx Y_{P1} \sin \omega t + Y_{Q1} \cos \omega t \\ y_N(t) &\approx \operatorname{Im}\left\{ \left(Y_{P1} + jY_{Q1} \right) e^{j\omega t} \right\} \end{aligned} \tag{1}$$

$$Y_{P1} = \frac{1}{\pi} \int_0^{2\pi} F\left(X_m \sin \omega t \right) \sin \omega t \, d(\omega t) \tag{2}$$

$$Y_{Q1} = \frac{1}{\pi} \int_0^{2\pi} F\left(X_m \sin \omega t \right) \cos \omega t \, d(\omega t) \tag{3}$$

where Y_{P1} and Y_{Q1} are first Fourier coefficients. Describing function is the ratio between first harmonic of the output signal and input signal in complex form:

$$G_N(X_m) = P(X_m) + jQ(X_m) = \frac{Y_{P1}}{X_m} + j\frac{Y_{Q1}}{X_m} \tag{4}$$

where $P(X_m)$ and $Q(X_m)$ are coefficients of the harmonic linearization [4, 7 and others]. Determination of describing function boils down to the determination of integral expressions (2, 3) for the known static characteristic of the nonlinear part of the system. If the static characteristic of nonlinear system cannot be analytically expressed or integral expressions can not be solved, describing function can be determined by experimental (simulation) method [1, 2] or by some method of numerical integration. In that case, the result is more often than not in form of graphical record of the experiment or simulation and the problem of determining the mathematical description of describing function arises. In the case of fuzzy systems, especially of Mamdani type, there is yet no good method for stability analysis. Use of describing function allows for such analysis by using well known and developed procedures ([4], [7] and others). In the case of fuzzy systems there was some work on analytical determination of describing function of fuzzy controllers. For example, analytical approach to determining fuzzy controller describing function is given in [9]. However, the obtained describing function is very complicated. Moreover, for the procedure to be applied a symmetrical fuzzy controller with triangular membership functions is required. In [1] an experimental method for the determination of the SISO fuzzy controller describing function by computer simulation is described. An experimental evaluation of the fuzzy controller describing function is also given in [8]. However, the

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 35–40.

procedure is not given in detail and the analysis is conducted for the system with the only one nonlinearity, thus only for a linear system controlled by the fuzzy controller. The method described in this work can be used to determine describing function of nonlinear systems with odd static characteristics, and is illustrated on Mamdani PD fuzzy element and Mamdani PD fuzzy element in series with saturation. The paper is organized as follows. In section II a method for approximation of static characteristic is given. Section III illustrates error estimation of approximation. Sections IV and V illustrate the procedure on example and Section VI gives the conclusion.

II. Determination of Describing Function Using Linearly Approximated Static Characteristic

Arbitrary nonlinear static characteristic can be approximated by the linear piecewise elements. Integral expressions (2, 3) for the describing function will be than expressed by an approximated analytical solution. General solution for the nonlinear odd static characteristic with hysteresis, approximated by the linear piecewise elements is presented in [3]. For the odd functions the analysis can be conducted in the first quadrant only. The approximation of the static characteristic $F_i(x)$ is shown in Fig. 2. Linearly approximated static characteristic is defined by two sets of points, rising and falling part of the static characteristic:

$$\left[g_i, F(g_i)\right],\ i = 1,\ldots,G \tag{5}$$

$$\left[h_i, F(h_i)\right],\ i = 1,\ldots,H \tag{6}$$

where set (5) defines rising part and set (6) defines falling part of the characteristic.

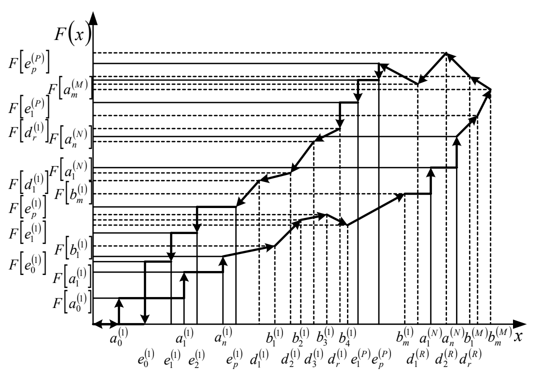


Fig.2 Static characteristic approximated with linear piecewise elements.

In order to simplify the presentation of the final expressions, sets (5) and (6) are further divided into groups of two types of linear elements with respect to the first derivative, step elements ($F'(x) = 0$ or ∞) and slope elements ($F'(x)\#0$ and finite). Each point of the set is designated with two indexes; the upper index denotes the order number of the group; the lower index denotes the order number of the element within the group. Rising part of static characteristic is defined by:

$$\left[a_i^{(j)}, F\left(a_i^{(j)}\right)\right],\ j = 1,\ldots,N,\ i = 1,\ldots,n(j) \tag{7}$$

$$\left[b_i^{(j)}, F\left(b_i^{(j)}\right)\right],\ j = 1,\ldots,M,\ i = 1,\ldots,m(j) \tag{8}$$

where pairs $\left[a_i^{(j)}, F\left(a_i^{(j)}\right)\right]$ denote position of step group points, pairs $\left[b_i^{(j)}, F\left(b_i^{(j)}\right)\right]$ denote position of slope group points, N is the number of step groups, $n(j)$ is the number of step elements per each group, M is the number of slope groups and $m(j)$ is the number of slope elements per each group. Falling part of the static characteristic is defined by:

$$\left[e_i^{(j)}, F\left(e_i^{(j)}\right)\right],\ j = 1,\ldots,P,\ i = 1,\ldots,p(j) \tag{9}$$

$$\left[d_i^{(j)}, F\left(d_i^{(j)}\right)\right],\ j = 1,\ldots,R,\ i = 1,\ldots,r(j) \tag{10}$$

where pairs $\left[e_i^{(j)}, F\left(e_i^{(j)}\right)\right]$ denote position of step group points, pairs $\left[d_i^{(j)}, F\left(d_i^{(j)}\right)\right]$ denote position of slope group points, P is the number of step groups, $p(j)$ is the number of step elements per each group, R is the number of slope groups and $r(j)$ is the number of slope elements per each group. Dividing the integral expressions (2, 3) into integral parts over the rising and falling parts of the static characteristic, and substituting integrals with the sum of integrals over the linear elements, we obtain the following algebraic expressions ([3]) of coefficients of the harmonic linearization:

$$P(X_m) = P_U(X_m) + P_S(X_m) \tag{11}$$

$$\begin{aligned}
P_U(X_m) = {} & \sum_{j=1}^{M}\Bigg\{\frac{K_m^{(j)}}{\pi} * \\
& * \Bigg(\arcsin\frac{b_m^{(j)}}{X_m} - \frac{b_m^{(j)} - 2b_{m-1}^{(j)}}{X_m}\sqrt{1-\frac{b_m^{(j)2}}{X_m^2}}\Bigg) - \\
& - \frac{K_1^{(j)}}{\pi}\Bigg(\arcsin\frac{b_0^{(j)}}{X_m} + \frac{b_0^{(j)}}{X_m}\sqrt{1-\frac{b_0^{(j)2}}{X_m^2}}\Bigg) - \\
& - \frac{2F\left(b_0^{(j)}\right)}{\pi X_m}\Bigg(\sqrt{1-\frac{b_m^{(j)2}}{X_m^2}} - \sqrt{1-\frac{b_0^{(j)2}}{X_m^2}}\Bigg) - \\
& - \frac{1}{\pi}\sum_{i=1}^{m-1}\Bigg[\left(K_{i+1}^{(j)} - K_i^{(j)}\right)\Bigg(\arcsin\frac{b_i^{(j)}}{X_m} + \frac{b_i^{(j)}}{X_m}\sqrt{1-\frac{b_i^{(j)2}}{X_m^2}}\Bigg) + \\
& + \frac{2K_i^{(j)}}{X_m}\left(b_i^{(j)} - b_{i-1}^{(j)}\right)\sqrt{1-\frac{b_m^{(j)2}}{X_m^2}}\Bigg]\Bigg\} + \\
& + \frac{2}{\pi X_m}\sum_{j=1}^{N}\Bigg\{F\left(a_0^{(j)}\right)\sqrt{1-\frac{a_0^{(j)2}}{X_m^2}} - F\left(a_n^{(j)}\right)\sqrt{1-\frac{a_n^{(j)2}}{X_m^2}} + \\
& + \sum_{i=1}^{n}\left(F\left(a_i^{(j)}\right) - F\left(a_{i-1}^{(j)}\right)\right)\sqrt{1-\frac{a_i^{(j)2}}{X_m^2}}\Bigg\}
\end{aligned} \tag{12}$$

$$P_S(X_m)=\sum_{j=1}^{R}\left\{\frac{L_r^{(j)}}{\pi}\left(\arcsin\frac{d_r^{(j)}}{X_m}+\frac{d_r^{(j)}-2d_{r-1}^{(j)}}{X_m}\sqrt{1-\frac{d_r^{(j)2}}{X_m^2}}\right)-\right.$$
$$-\frac{L_1^{(j)}}{\pi}\left(\arcsin\frac{d_0^{(j)}}{X_m}+\frac{d_0^{(j)}}{X_m}\sqrt{1-\frac{d_0^{(j)2}}{X_m^2}}\right)-$$
$$-\frac{2F\left(d_0^{(j)}\right)}{X_m}\left(\sqrt{1-\frac{d_r^{(j)2}}{X_m^2}}-\sqrt{1-\frac{d_0^{(j)2}}{X_m^2}}\right)-$$
$$-\frac{1}{\pi}\sum_{i=1}^{r-1}\left[\left(L_{i+1}^{(j)}-L_i^{(j)}\right)\left(\arcsin\frac{d_i^{(j)}}{X_m}+\frac{d_i^{(j)}}{X_m}\sqrt{1-\frac{d_i^{(j)2}}{X_m^2}}\right)+\right.$$
$$\left.\left.+\frac{2L_i^{(j)}}{X_m}\left(d_i^{(j)}-d_{i-1}^{(j)}\right)\sqrt{1-\frac{d_r^{(j)2}}{X_m^2}}\right]\right\}+$$
$$+\frac{2}{\pi X_m}\sum_{j=1}^{P}\left\{F\left(e_0^{(j)}\right)\sqrt{1-\frac{e_0^{(j)2}}{X_m^2}}-F\left(e_p^{(j)}\right)\sqrt{1-\frac{e_p^{(j)2}}{X_m^2}}+\right.$$
$$\left.+\sum_{i=1}^{p}\left(F\left(e_i^{(j)}\right)-F\left(e_{i-1}^{(j)}\right)\right)\sqrt{1-\frac{e_i^{(j)2}}{X_m^2}}\right\} \qquad (13)$$

$$Q(X_m)=Q_U(X_m)+Q_S(X_m) \qquad (14)$$

$$Q_U(X_m)=\sum_{j=1}^{M}\left\{\frac{2}{\pi X_m}F\left(b_0^{(j)}\right)\left(\frac{b_m^{(j)}}{X_m}-\frac{b_0^{(j)}}{X_m}\right)+\right.$$
$$+\frac{K_m^{(j)}b_m^{(j)2}}{\pi X_m^2}+\frac{K_1^{(j)}b_0^{(j)}}{\pi X_m}\left(\frac{b_0^{(j)}}{X_m}-2\frac{b_m^{(j)}}{X_m}\right)+$$
$$\left.+\sum_{i=1}^{m-1}\left(K_{i+1}^{(j)}-K_i^{(j)}\right)\frac{b_i^{(j)}}{\pi X_m}\left(\frac{b_i^{(j)}}{X_m}-2\frac{b_m^{(j)}}{X_m}\right)\right\}+$$
$$+\frac{2}{\pi X_m}\sum_{j=1}^{N}\left\{F\left(a_n^{(j)}\right)\frac{a_n^{(j)}}{X_m}-F\left(a_0^{(j)}\right)\frac{a_0^{(j)}}{X_m}-\right.$$
$$\left.-\sum_{i=1}^{n}\left[F\left(a_i^{(j)}\right)-F\left(a_{i-1}^{(j)}\right)\right]\frac{a_i^{(j)}}{X_m}\right\} \qquad (15)$$

$$Q_S(X_m)=-\sum_{j=1}^{R}\left\{\frac{2}{\pi X_m}F\left(d_0^{(j)}\right)\left(\frac{d_r^{(j)}}{X_m}-\frac{d_0^{(j)}}{X_m}\right)+\right.$$
$$+\frac{L_r^{(j)}d_r^{(j)2}}{\pi X_m^2}+\frac{L_1^{(j)}d_0^{(j)}}{\pi X_m}\left(\frac{d_0^{(j)}}{X_m}-2\frac{d_r^{(j)}}{X_m}\right)+$$
$$\left.+\sum_{i=1}^{r-1}\left(L_{i+1}^{(j)}-L_i^{(j)}\right)\frac{d_i^{(j)}}{\pi X_m}\left(\frac{d_i^{(j)}}{X_m}-2\frac{d_r^{(j)}}{X_m}\right)\right\}-$$
$$-\frac{2}{\pi X_m}\sum_{j=1}^{P}\left\{F\left(e_p^{(j)}\right)\frac{e_p^{(j)}}{X_m}-F\left(e_0^{(j)}\right)\frac{e_0^{(j)}}{X_m}-\right.$$
$$\left.-\sum_{i=1}^{p}\left[F\left(e_i^{(j)}\right)-F\left(e_{i-1}^{(j)}\right)\right]\frac{e_i^{(j)}}{X_m}\right\} \qquad (16)$$

where $P_U(X_m)$ and $Q_U(X_m)$ are coefficients of the harmonic linearization over rising characteristic, and $P_S(X_m)$ and $Q_S(X_m)$ are coefficients of the harmonic linearization over falling characteristic.

III. Error Estimation

Determination of describing function using linearly approximated static characteristic introduces an inherent error into approximated describing functions. The error is the result of the difference between the original and linearly approximated static characteristics. Approximation of an arbitrary function with piecewise polynomial functions is commonly known as Lagrange interpolation [6]. Lagrange interpolation polynomial of the first order, or the linear Lagrange polynomial can be written as:

$$p_1(x)=\frac{x-x_1}{x_0-x_1}f(x_0)+\frac{x-x_0}{x_1-x_0}f(x_1) \qquad (17)$$

where $\lfloor x_0,f(x_0)\rfloor$ and $\lfloor x_1,f(x_1)\rfloor$ are coordinates of the end points of linear element. It can be showed ([6]) that a close estimate of the error of approximation with linear Lagrange polynomial can be expressed as:

$$e_1(x)=(x-x_0)(x-x_1)\frac{f''(x)}{2} \qquad (18)$$

Substituting the smallest and the largest values of the second derivative of the function $f''(x)$ on the interval $x_0 \leqslant x \leqslant x_1$, we obtain boundaries of the error on the interval. The second derivative of the function $f''(x)$, in the expression (18) also has to be approximated using available sets of interpolation points. Approximate of the second derivative of the function can be derived by deriving the Lagrange interpolation polynomial of the higher order. Second derivative of the Lagrange interpolation polynomial of the second order, defined by the three points $\lfloor x_0,f(x_0)\rfloor$, $\lfloor x_1,f(x_1)\rfloor$ and $\lfloor x_2,f(x_2)\rfloor$, on the interval $x_0 \leqslant x \leqslant x_2$, can be written as ([6]):

$$f''(x)\approx p_2''(x)=\frac{2f(x_0)}{(x_1-x_0)(x_2-x_0)}-$$
$$-\frac{2f(x_1)}{(x_1-x_0)(x_2-x_1)}+\frac{2f(x_2)}{(x_2-x_0)(x_2-x_1)} \qquad (19)$$

The second derivative approximation (19) is constant on the interval $x_0 \leqslant x \leqslant x_2$ and represents average value of the second derivative of the function on the given interval.

Static characteristic of the nonlinear element $F(x)$, including error estimate function (18), is then the sum of linearly approximated static characteristic $F_i(x)$ and corresponding error estimate function:

$$F(x)=F_i(x)+E_F(x) \qquad (20)$$

Output signal $y_N(t)$ is thus sum of linearly approximated static characteristic and corresponding error estimate function:

$$y_N(t)=F(x)=F_i(x)+E_F(x) \qquad (21)$$

By substituting expression (21) into integral expressions for coefficients of the harmonic linearization $P(X_m)$ and $Q(X_m)$ (2, 3, 4), we can separate integrals into sum of integrals over linearly approximated static characteristic and corresponding error estimate function. Integral expressions over linearly approximated static characteristic represent describing function determined using linearly approximated static characteristic, given in [3] (11 – 16). Integral expressions over corresponding error estimate function represent error estimation of the approximated describing function. Integral expressions for error estimation of the determination of the coefficients of the harmonic linearization can thus be written as:

$$E_P(X_m) = \frac{1}{\pi X_m}\int_0^{2\pi} E_F(X_m \sin\omega t)\sin\omega t\, d(\omega t) \quad (22)$$

$$E_Q(X_m) = \frac{1}{\pi X_m}\int_0^{2\pi} E_F(X_m \sin\omega t)\cos\omega t\, d(\omega t) \quad (23)$$

where $E_P(X_m)$ denotes error estimation of $P(X_m)$ and $E_Q(X_m)$ denotes error estimation of $Q(X_m)$. By substituting integrals in the equations (22, 23) with the sum of integrals over the linear elements, and performing algebraic reductions, we get the final algebraic expressions for error estimation:

$$\begin{aligned}
&E_P(X_m) = \\
&\frac{X_m}{\pi}\sum_{i=1}^{G} E_{FU\,i}{}'' \left[\frac{g_{i-1}+g_i}{2X_m}\left(\arcsin\frac{g_{i-1}}{X_m} - \arcsin\frac{g_i}{X_m}\right) + \right.\\
&+\left(\frac{2}{3}+\frac{g_{i-1}g_i}{2X_m^2}\right)\left(\sqrt{1-\frac{g_{i-1}^2}{X_m^2}} - \sqrt{1-\frac{g_i^2}{X_m^2}}\right) - \\
&\left. -\frac{1}{6}\left(\frac{g_{i-1}^2}{X_m^2}\sqrt{1-\frac{g_{i-1}^2}{X_m^2}} - \frac{g_i^2}{X_m^2}\sqrt{1-\frac{g_i^2}{X_m^2}}\right)\right] + \\
&+\frac{X_m}{\pi}\sum_{i=1}^{H} E_{FS\,i}{}'' \left[\frac{h_{i\ 1}+h_i}{2X_m}\left(\arcsin\frac{h_{i-1}}{X_m} - \arcsin\frac{h_i}{X_m}\right) + \right.\\
&+\left(\frac{2}{3}+\frac{h_{i-1}h_i}{2X_m^2}\right)\left(\sqrt{1-\frac{h_{i-1}^2}{X_m^2}} - \sqrt{1-\frac{h_i^2}{X_m^2}}\right) - \\
&\left. -\frac{1}{6}\left(\frac{h_{i-1}^2}{X_m^2}\sqrt{1-\frac{h_{i-1}^2}{X_m^2}} - \frac{h_i^2}{X_m^2}\sqrt{1-\frac{h_i^2}{X_m^2}}\right)\right]
\end{aligned} \quad (24)$$

$$\begin{aligned}
&E_Q(X_m) = \frac{X_m}{\pi}\sum_{i=1}^{G} E_{FU\,i}{}'' * \\
&*\left[\frac{g_{i-1}^2}{X_m^2}\left(\frac{1}{6}\frac{g_{i-1}}{X_m} - \frac{1}{2}\frac{g_i}{X_m}\right) - \frac{g_i^2}{X_m^2}\left(\frac{1}{6}\frac{g_i}{X_m} - \frac{1}{2}\frac{g_{i-1}}{X_m}\right)\right] + \\
&+\frac{X_m}{\pi}\sum_{i=1}^{H} E_{FS\,i}{}'' * \\
&*\left[\frac{h_{i-1}^2}{X_m^2}\left(\frac{1}{6}\frac{h_{i-1}}{X_m} - \frac{1}{2}\frac{h_i}{X_m}\right) - \frac{h_i^2}{X_m^2}\left(\frac{1}{6}\frac{h_i}{X_m} - \frac{1}{2}\frac{h_{i-1}}{X_m}\right)\right]
\end{aligned} \quad (25)$$

IV. Example: Describing Function and Error Estimation of Fuzzy Element

The procedure of determination of describing function using linearly approximated describing function and estimation of the error of resulting describing function is illustrated here on the Mamdani fuzzy nonlinear element, using static characteristic in the region of small frequencies.

The block diagram of the fuzzy element is shown in the Fig. 3. Membership functions are shown in the Fig. 4, and rulebase table is shown in the Fig. 5. Values of the proportional and derivative gains used in this example are k_p=1.2 and k_d=0.001.

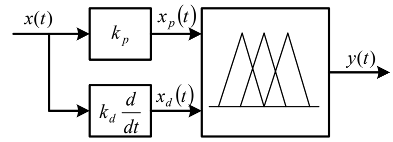


Fig.3 The block diagram of the fuzzy element.

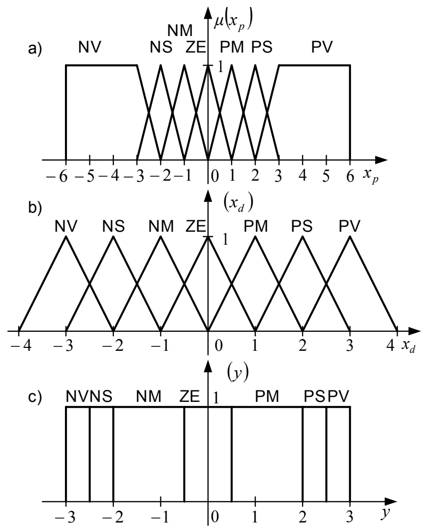


Fig.4 Membership functions of the fuzzy element: a) proportional input, b) derivatve input and c) output.

x_d \ x_p	**NV**	**NS**	**NM**	**ZE**	**PM**	**PS**	**PV**
NV	NV	NV	NV	NV	NS	NM	ZE
NS	NV	NV	NV	NS	NM	ZE	PM
NM	NV	NV	NS	NM	ZE	PM	PS
ZE	NV	NS	NM	ZE	PM	PS	PV
PM	NS	NM	ZE	PM	PS	PV	PV
PS	NM	ZE	PM	PS	PV	PV	PV
PV	ZE	PM	PS	PV	PV	PV	PV

Fig.5 Rulebase of the fuzzy element.

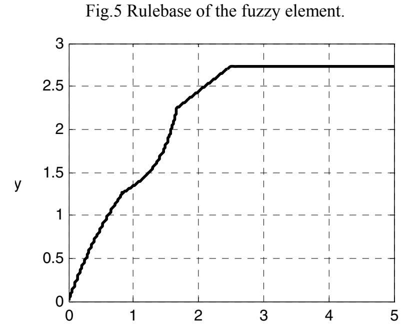


Fig.6 Static characteristic of the fuzzy element in the region of small frequencies.

Static characteristic in the region of small frequencies is shown in the Fig 6. Describing function determined using linearly approximated static characteristic (using linearization interval $\delta x = 0.05$) is shown in the Fig. 7. Corresponding error estimation of determined describing function is shown in the Fig. 8. The difference between describing functions determined experimentally [1, 2] and using linearly approximated static characteristic is shown in the Fig. 9.

Values of the error estimation are, in the large region, somewhat smaller than $1*10^{-3}$. Compared to the values of the describing function, these are three orders of magnitude smaller values, which is very satisfactory result that justifies application of the method in the determination of describing function of fuzzy element. As an additional test, determined describing function is compared to the describing function determined experimentally (such describing function of course is not available in analytic form, only as a graph, and thus not suitable for system analysis purposes). The difference between these two describing functions (Fig.9) is, in the large part, around 1%, with maximal difference of 3%. Considerably larger values of the difference between these describing functions, comparing to the values of error estimation, are largely due to the error of describing function detemined experimentally. These values further justify the use of the method.

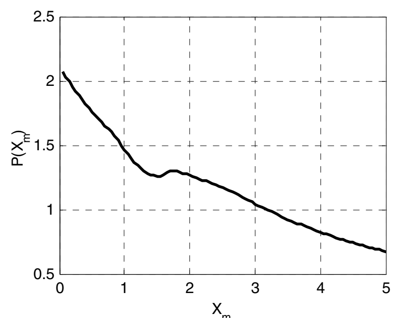


Fig.7 Describing function of the fuzzy element.

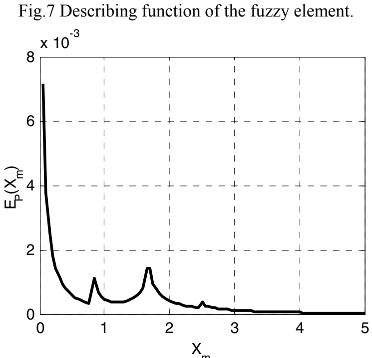


Fig.8 Error estimation of describing function of the fuzzy element.

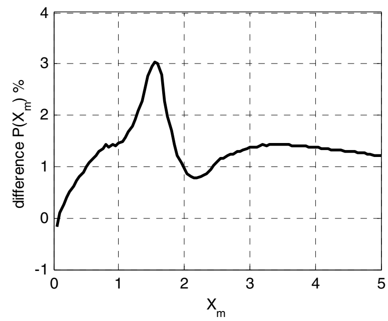


Fig.9 Diffference between descibing functions determined experimentally and using linearly approximated static characteristic.

V. Example: Describing Function and Error Estimation of Nonlinear System with Fuzzy Element and Saturation Nonlinearity

The procedure is here illustrated on the example of nonlinear system with fuzzy element and saturation nonlinear element. The block diagram of the nonlinear system is shown in the Fig 10. Fuzzy element used in this example is the same as used in previous example. In this example we use saturation nonlinearity with saturation boundary at d=2. Static characteristic of nonlinear system is shown in the Fig. 11. In the first part, up to the level of output signal y=2, static characteristic is the same as static characteristic of the fuzzy element, and at that point it enters the region of saturation.

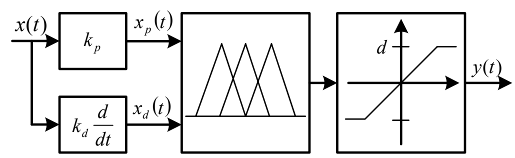


Fig.10 The block diagram of nonliner system with fuzzy element and saturation nonlinearity.

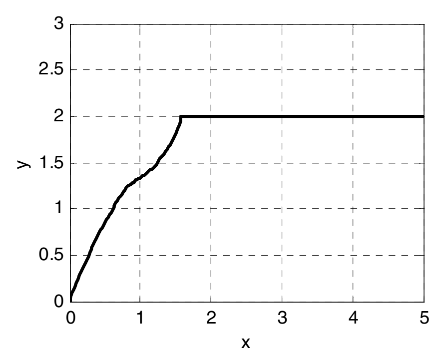


Fig.11 Static characteristic of the nonlinear system with fuzzy element and saturation nonlinerity with saturation boundary d=2.

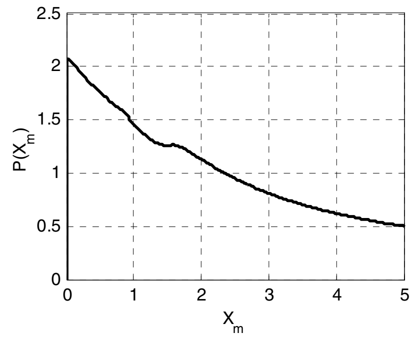


Fig.12 Describing function of the nonlinear system with fuzzy element and saturation nonlinerity with saturation boundary *d*=2.

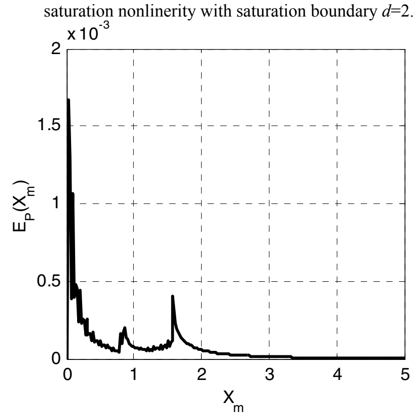


Fig.13 Error estimation of the describing function of the nonlinear system with fuzzy element and saturation nonlinerity with saturation boundary *d*=2.

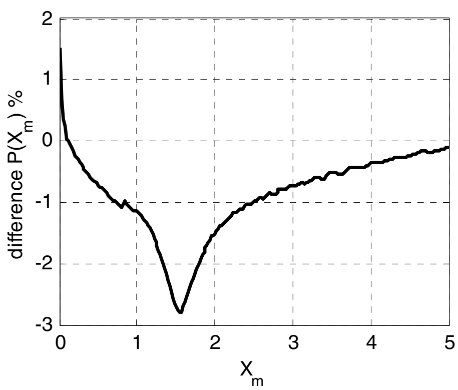


Fig.14 Diffference between descibing functions determined experimentally and using lineary approximated static characteristic.

Describing function determined using lineary approximated static characteristic (using linearization interval $\delta x = 0.05$) is shown in the Fig. 12. Corresponding error estimation of determined describing function is shown in the Fig. 13. The difference between describing functions determined experimentally [1, 2] and using linearly approximated static characteristic is shown in the Fig. 14. Like in the previous example, error estimation is approximately three orders of magnitude lower than the values of describing function. The difference between describing function determined experimetally and determined using linearly approximated static characteristic is in this example also around 1%, with maximal value below 3%.

VI. CONCLUSION

Generalized procedure for approximate analytical determination of describing function and corresponding error estimation of nonlinear systems with odd static characteristic with hysteresis is given. The procedure is employed in determination of describing function of nonlinear systems with fuzzy element. Presented results show high level of accuracy of gained approximate describing functions. The method is convenient for determination of describing functions of nonlinear systems for which analytical description is not known, like fuzzy elements. With data acquisition software and hardware tools available today the experiment to obtain static characteristic is easily setup and graphical representation of static characteristics can be obtained relatively simply. It is shown that the estimation error of describing function is small, less than 3%. This precision is satisfactory in engineering practice to determine characteristics and stability of controlled system, especially when dealing with filter like systems.

REFERENCES

[1] O.Kuljača, S.Tešnjak and Z.Vukić, *"Describing Function of Mamdani Type Fuzzy Controller with Input Signals Derived From Single System Input and Singleton Ouput Membership Functions"*, Proceedings of the 1999 IEEE Hong Kong Symposium on Robotics and Control, Volume I, HKSRC '99, Hong Kong, 1999, pp. 327-331.

[2] T.Šijak, O.Kuljača, and S.Tešnjak, *"Stability analysis of fuzzy control system using describing function method"*, MED'01, Dubrovnik Croatia, 26-29.06.2001

[3] T.Sijak, *"Describing Function Method for Analysis of Nonlinear Control Systems with Fuzzy Logic Elements"*, Sveuciliste u Zagrebu, Zagreb 2006, (in Croatian)

[4] Z.Vukic, Lj.Kuljaca, D.Donlagic, S.Tesnjak, "*Nonlinear Control Systems*", Marcel Dekker, New York, Basel 2003

[5] Zoran Vukic, Ljubomir Kuljaca, "*Automatic Control – analysis of linear systems*", Kigen, Zagreb, 2004 (in Croatian)

[6] Erwin Kreyszig, *"Advanced Engineering Mathematics"*, Seventh Edition, John Wiley and Sons, 1993

[7] J.J.E. Slotine, W. Li, *"Applied Nonlinear Control"*, Prentice Hall, 1991

[8] F. Gordillo, J. Aracil and T. Alamo, "Determing Limit Cycles in Fuzzy Control Systems", FUZZ-IEEE'97, Barcelona, 193-198, 1997

[9] E. Kim, H. Lee, M. Park, Limit-Cycle Prediction of a Fuzzy Control System Based on Describing Function Method, IEEE Transactions on Fuzzy Systems, Vol. 8, No 1., February 2000, pp. 11-21, 2000

Semantic Object Generation in Tongue Image Analysis

XIE Zhiming，YANG Bao-an , CHAI xin

School of Management, Donghua University, P.R.China, 200051

Abstract: A method of computerized tongue image analysis based on image processing for the purpose of digitizing the tongue properties in traditional Chinese medical diagnosis is presented. A hybrid method which uses Support Vector Machine to extract the semantic object, and a combination kernal function is selection after many compare. Finite Mixture Model and many image process methods is applied into diagnosis system. The experiment of the system shows that methods proposed are effective. The following results are presented in the article:

1) A multiply semantic image model is built our literature, which contributes abundant character to determine disease.

2) The SVM classifications are applied to transaction from the lower level to the top ones. The complex of the SVM classifications depends on the sample number rather than the characteristic dimension, which can satisfy the requirement of the system.

3) An application implements the approaches mentioned by the literature is introduced, through which the effect of the model are proved.

Keywords: Tongue image; Support Vector Machine; Finite Mixture Model; Image Semantic Model; Chinese medical diagnosis

I. INTRODUCTION

The main information sources for Chinese traditional medicine consist of "look", "smell", "inquiry" and "feel". As the important part of "look" means, tongue diagnosis is significant to catch the patient's status. Tongue diagnosis depends on the tongue image basically and scientific classification for Chinese traditional tongue diagnosis approaches is crucial for modernization of Chinese traditional medicine.

The existing methods always infer by rule, such as, ZHOU Yue, et al [18] partition some characters and set the threshold as condition to identify the result. The traditional approaches have the following defects:

1) The rule only can represent a little of Chinese traditional medicine knowledge. As we all know, the modernization for Chinese traditional medicine is still on process and therefore the work to import all the diagnosis knowledge into the reasoning rules completely is impossible.

2) A case can be seen as a complex object that contains some problems cannot be represented by simple conditions. The partitions and the thresholds are somewhere dogmatic.

3) The rule knowledge can hardly be transformed to other form, which is required by modern medicine. Maturity system concludes the disease by combination of many techniques. Hence, the knowledge in analysis system need to be extensive.

The problems mentioned aforementioned can be solved successfully by the extraction the semantic object from image. Firstly, semantic information is some description for tongue image and has much more detail than the rule. Furthermore, semantic reasoning is similarity to human being knowledge, which are recorded on books, paper and other documents. Lastly, semantic information can be transformed into other form easily.

In a nutshell, the procedure of tongue diagnosis can be simplified as a kind of image process which extracts the disease feature from the picture of tongue. In the paper, a hybrid method will be introduced and the experiments will show its good consequence.

II. IMAGE SEMANTIC MODEL OVERVIEW

A. *Introduction of the Model*

In order to make clear the process how to transfer the image into the illness information, author establish the model composed of four levels: feature level, object

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 41–46.

level, conception level and diagnosis level. The feature level always handles the content-based character, such as color, shape, texture and area. The object level mainly concerns the problems of objects and the relationship between objects. The conception level will synthesize the object level and present the meanings, behaves of the images. The highest level, diagnosis level will purify the conception level and show the information which the doctors need.

Definition (Image Semantic Model) Image Semantic Model M is the description for advantage image knowledge structure, which extracts the image feature $\mathrm{A}(f_1, f_2, \cdots f_\mathrm{n})$ and uses a structure method to represent the relationship between the property. Hence,

$$M : D(\overline{f_1, f_2, \cdots f_\mathrm{n}}, KD, OP) \Rightarrow knowlege\_chains$$

Where KD is the knowledge base and OP is the operations and rules to reason. The knowledge chains are what we need. In this paper, it stands for the some notations in Chinese traditional medicine.

To work out a computerized model, the article use the FMM(Finite Mixture Model)[1] to evaluate the classification distribution of the conception objects. The conception C_j classification probability is:

$$P(X, C_j, k, \omega_{C_j}, \theta_{C_j}) = \sum_{i=1}^{k} P(X \mid S_i, \theta_{S_i}) \varpi_{S_i}$$

Where: C is the conception, S is the semantic and $P(X \mid S_i, \theta_{S_i})$ is the probability of i-th multi-dimensions mix component，k is the selected component ， $\theta_{C_j} = \{\theta_{S_j}, i = 1,2,\cdots k\}$ is the multi-dimensions parameter of model 。 $\varpi_{\mathrm{C}_i} = \{\varpi_{S_i}, i = 1,2,\cdots k\}$ is the weight vector for multi-dimensions mix component ， X is a multi-dimensions object vector.

B. Image Semantic Classification by Using SVM

Image classification and clustering include supervised and unsupervised classification of images[1]. In the supervised classification supervised classification, we are given a collection of labeled images (or priori knowledge), and the problem is to label a newly encountered, yet unlabeled image. On the other hand, for unsupervised classification (or image clustering), the problem is to group a given collection of unlabeled images into meaningful clusters according to the image visual feature without a prior knowledge[2].

We have tried to cluster images into semantic categories using SOFM (Self-organism Feature Mapping) and C-Means. The experiment results show that the error rates are very high. The reason perhaps is that the low-level visual features are not related to the human perception about image content. The clustering algorithm couldn't automatically bridge the enormous gap between low-level visual feature and high-level semantic content without priori knowledge.

We believe supervised classification is a promising method, and there have been many achievements in this field. Smith[3] proposed a multi-stage image classification algorithm based on visual features and related text. Bruzzone and Prieto[4] developed a variety of classifiers to label the pixels in a landset multi-spectral scanner image. MM-classifier developed by Zaiane et al.[5] classifies multimedia images based on some provided class labels. Vailaya et al.[6] used a binary Bayesian classifier for the hierarchical classification of vacation images into indoor/outdoor classes. Outdoor images are further classified into city/landscape classes. Li and Wang[7] classified textured and non-textured images using region segmentation. Experiments performed by Chapelle[8] have shown that SVM could generalize well to image classification, however the only features are high-dimension color histogram, which simply quantizes RGB color space into 4096 color bins. In fact, image content cannot be represented effectively by only color feature. For example, lawn and trees may have the same color histogram, while lawns in spring and autumn have different color histograms whilst they have the same shape or semantic feature.

Therefore, we have to find an efficient method to describe the image content and bridge the gap between the low-level visual feature and high-level semantic information. Taking these into consideration, we propose more effective texture and edge descriptors in this paper. Based on this, by combining color, texture and edge features seamlessly, images are grouped into semantic categories using SVM.

Support vector machine is a well-known pattern classification method. It is an approximate implementation of the structural risk minimization (SRM) principle[9] and creates a classifier with minimized Vapnik-Chervonenkis (VC) dimension. For pattern classification, SVM has a very good generalization performance without domain knowledge of the problems. This is one of the reasons why we select SVM as an image classifier.

Let the separable training set be $\{(\vec{x}_i, y_i)\}_{i=1}^{N}$ where $\vec{x}_i$ is the input pattern vector, $y_i \in \{-1,1\}$ is the class label, +1 denotes the positive example, and -1 denotes the negative example. If the training set is linearly separable, and the discriminating function is g(x) = $w^T x + b$, we can easily get the classifier hyper-plane by calculating: $\overline{w}^T \overline{x} + b = 0$, where $\overline{w}$ is a weight vector, b is a bias. The goal of the SVM is to find the parameters w_0 and b_0 such that the distance between the hyper-plane and the nearest sample point is more than 1:

$$\overline{w_0}^T \overline{x} + b_0 \geq 1 \quad \text{for} \quad y_i = +1$$

$$\overline{w_0}^T \overline{x} + b_0 \leq 1 \quad \text{for} \quad y_i = -1$$

If it is not linearly separable, it is necessary to map the input training vector into a high-dimension feature space using a kernel function $K(\vec{x}, \vec{x}_i)$, then create the optimal hyper-plane and implement the classification in the high-dimension feature space. For more details, refer to Ref.[10].

At this point，we have tw o sets of kernels：content kemel K_C that is calculated by using visual and textual features， and linkage kernel K_L (i.e. $K_L^{(D)}$ or $K_L^{(C)}$). We can combine the tw o kernels to obtain a valid kemel that can perform better than the other two considered separately. Joachims [11] validated that the combination of kernels is beneficial as long as both kernels are independent in that they do not extract the same features.

The simplest method is the convex combination of the content kemel K_C and the linkage kemel K_L, i.e.,

$$K = (1-\beta)K_C + \beta K_L$$

where β is a weight，$0 \leq \beta \leq 1$. Altematively，Kandola et a1[12] proposed a von Neumann kernel，based on the mutual reinforcement assumption. The von Neumann kemel can be treated as a non-1inear combination of two kernels. However，this method also suffers from much higher complexity due to the iterative computation. We thus do not intend to apply it in this Paper.

Similarly to the topographic SVM in [13]，we propose a RSVC model using the composite kernel for collective classification. If the vector α and the scalar b are the parameters of the hyperplane learned from the training images，then the RSVC decision rule is defined as：

$$\begin{aligned} y_i &= \operatorname{sgn}\Big(\sum_{j=1}^{l} \alpha_j y_j [(1-\beta)K_C(f_i^{(C)}, f_j^{(C)}) \\ &\quad + \beta K_L(f_i^{(L)}, f_j^{(L)})] + b\Big) \\ &= \operatorname{sgn}\Big(\beta \sum_{j=1}^{l} \alpha_j y_j K_C(f_i^{(L)}, f_j^{(L)}) + \theta_i\Big) \end{aligned}$$

Where $\theta_i = (1-\beta)\sum_{j=1}^{l} \alpha_j y_j K_C(f_i^C, f_j^C) + b$ which is the decision function of a conventional SVM，. And I denotes the number of support vectors(SVs). Here each image is represented by $(<f_i^C, f_j^C>, y_i)$， where f_i^C and f_j^C are its content feature vector and link feature vector respectively，y_i is its class labe1. Note that when it is trained using the composite kernel，the resulting SVs will still contain the relevant inform ation about the content features， and the link feature information required good distinction of the classes.

However，the situation is different when we use $K_L^{(C)}$ as the link age kernels. As mentioned before，the calculation of the link features needs the neighboring kernel label attributes of $I_i \in I_{(T)}$. Thus to collectively classify the images in $I_{(T)}$, an iterative approach is used to achieve a self-consistent solution to the classification problem. W e denote the label at step τ as $y_i|_\tau$, and use $\widetilde{f}_i^{(L)}|_r$ to denote the link feature at step τ. Then at each step τ new labels are assigned according to

$$y_i|_\tau = \mathrm{sgn}(\beta \sum_{j=1}^{l} \alpha_j y_j K_L^{(C)}(\widetilde{f}_i^{(L)}|_{\tau-1}, \widetilde{f}_j^{(L)}|_{\tau-1}) + \theta_i$$

where $y_i|_{\tau=0} = \mathrm{sgn}(\theta_i)$ and θ_i do not change with τ. This leads to an iterative assignment of new labels: at step $\tau = 0$. The results from a conventional SVM $\beta = 0$ are used to initialize the labels; at the following steps, the estimates of the neighboring labels are available from the previous iteration. And a criterion may be used to determine whether the iteration process will be terminat, i.e $y_i|_\tau = y_i|_{\tau-1}, \forall I_i \in I_{(T)}$ with a minimal τ.

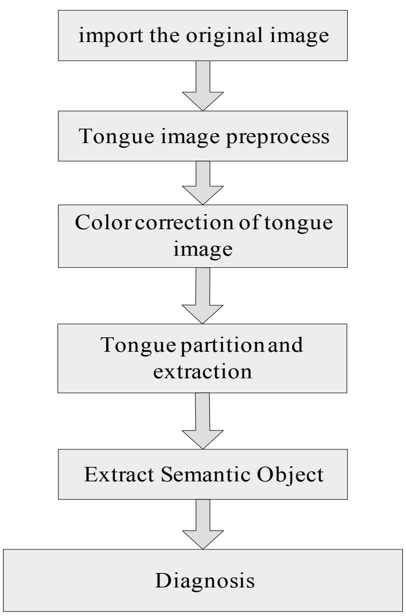


Fig1 Diagnosis Procedure

III. Diagnosis System

The figure 1 describes the process that our system diagnoses the disease information from the tongue images.

A. Tongue image preprocess

Preprocess section includes general image process algorithm, such as RGB color adjustment, brightness/contrast adjustment, rotation, zoom in/out transform, sharpen, blur and edge enhancement. After these steps, we can complete some basic function, de-noise and filter, which can help us get more correct outcome.

B. Color Correction

This project use the self-defined color scale for correction. Generate speaking, the more color scale are, the correction effect will be better. However, in this project, we only need focus some color because the tongue color exists in a special range.

Fig2 the color scale in this project

From the figure 2, we can find the diagnosis device mainly deal with the red and green color, which are the general color of human tongue.

The figure3 and figure 4 present the diagnosis effect.

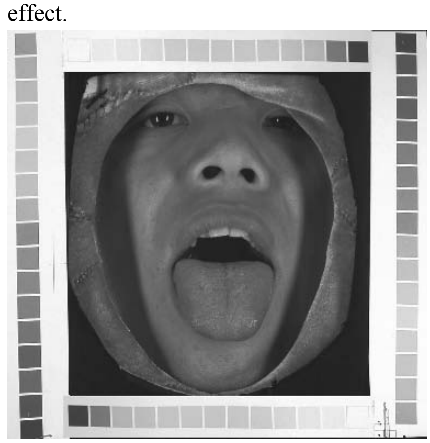


Fig 3 original image

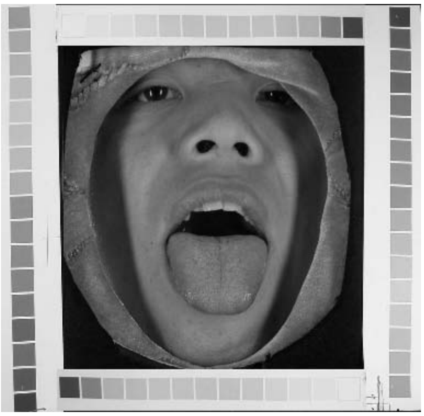


Fig 4 After correction

C. *Tongue partition and extraction*

According to the Chinese traditional medicine theory, different part of tongue corresponds with different body. Therefore it exists the requirement to divide the tongue automatically.

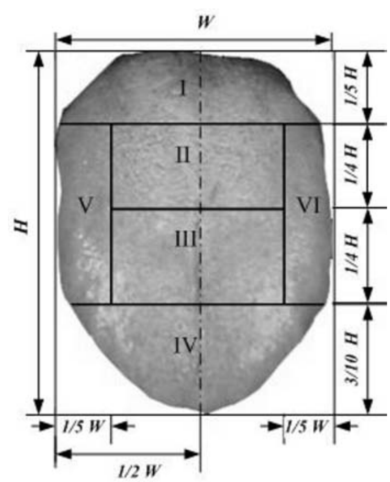


Fig5 Five partitions of tongue corresponding to health of different apparatus.

In the above figure, Part V and VI represent the health of liver and bladder, the part II and III provide the information of stomach and spleen. Part I corresponds to kidney and Part IV corresponds with heart and lung.

We use the JSEG algorithm to partition the tongue images. The figure 6 show how to divide tongue into four parts by JSEG.

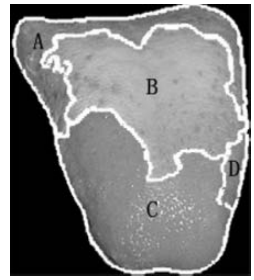


Fig 6 JSEG Algorithm

D. *Extract Semantic Object*

The methed mentioned in section II will be applied to extract semantic object.

Figure 7 presents the stucture of the semantic extraction process.

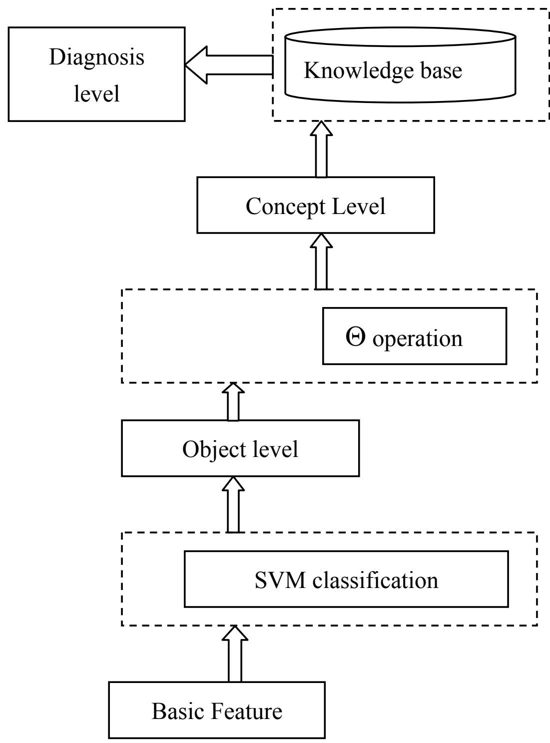


Fig 7 Semantic Model Structure

IV. Summary and Outlook

We explore the tongue images analysis by extraction the semantic information. The four level image semantic models are built in our literature: feature level, object level, conception level and diagnosis level.

Different level knowledge contributes abundant character to determine disease. The SVM classifications are applied to transaction from the lower level to the top ones. The complex of the SVM classifications depends on the sample number rather than the characteristic dimension, which can be accepted by the system. Last but not the least, the semantic object can be combined with other system information to deduce.

REFERENCES

[1] Jianping Fan, Yuli Gao, Hangzai Luo and Guangyou Xu. Statisitical modeling and conceptualization of natural images[J]. Pattern Recognition, 2005, 38(2005): 865-885

[2] Jain AK, Murty MN, Flynn PJ. Data clustering: A review. ACM Computing Survey, 1999,31(3):264~323.

[3] Smith JR, Chang SF. Multi-Stage classification of images from features and related text. In: Proceedings of the 4th Europe EDLOS Workshop. San Miniato, 1997. http://www.ctr.columbia.edu/~jrsmith/html/pubs/DELOS97/delosfin/delosfin.html.

[4] Bruzzone L, Prieto DF. Unsupervised retraining of a maximum likelihood classifier for the analysis of multitemporal remote sensing images. IEEE Transactions on Geoscience and Remote Sensing, 2001,39(2):456~460.

[5] Zaiane OR. Resource and knowledge discovery from the Internet and multimedia repositories [Ph.D. Thesis]. Burnaby, Simon Fraser University, 1999.

[6] Vailaya A, Figueiredo AT, Jain AK, Zhang HJ. Image classification for content-based indexing. IEEE Transactions on Image Processing, 2001,10(1):117~130.

[7] Li J, Wang JZ, Wiederhold G. Classification of textured and non-textured images using region segmentation. In: Proceedings of the 7th International Conference on Image Processing. Vancouver, 2000. 754~757.

[8] Chapelle O, Haffner P, Vapnik V. SVMs for histogram-based image classification. IEEE Transactions on Neural Network, 1999,9. http://citeseer.nj.nec.com/chapelle99svms.html

[9] Pengyu Hong, Qi Tian, et al. Incorporate support vector machines to content-based image retrieval with relevant feedback. In: Proceedings of the International Conference Image Processing. 2000. 750~753.

[10] Vapnik VN, translated by Zhang XG. The Nature of Statistical Learning Theory. Beijing: Tsinghua University Press, 2000

[11] Joachims. T， Cristianini, N., Shawe． Talyor, J, 2001．Composite Kemels for Hypertext Categorization．Proceedings of the Intemational Conference on Machine Learning(ICML•2001)，Morgan Kaunfmann Publishers，San Francisco。US．P．250．257．

[12] Kandola.J Shawe Talyor，J ，Cristianini，N ，2002．Learning Seman tic Similarity． Proceedings of Intemational Conference on Advances in Information Processing System(NIPS)．

[13] Mohr,J， Obermaye.K 2005． A Topographic Suppo~Vector Machine：Classification Using Local Label Configurations Advances in Neural Information Processing Systems l7．MIT Press，Cambridge，MA，P929-936．

[14] M.E.Yahia, R.Mahmod, N.Sulaiman, etc. Rough neural expert systems. Expert Systems with Applications , 2000, Vol.18 , pp:87-99.

[15] William P.Wagner, Mohammad K.Najdawi, Q.B.Chung. Selection of knowledge acquisition techniques based upon the problem domain characteristics of production and operations management expert systems. Expert Systems, May 2001, Vol.18, pp.312-345

[16] Nigel Shadbolt, Kieron O'Hara, Louise Crow. The experimental evaluation of knowledge acquisition techniques and methods: history, problems and new directions. International Journal of Human-Computer Studies, 1999, Vol. 51, pp:729-755.

[17] Buchanan B G,Constructing an expert system, Building Expert Systems, Chapter5. Reading, M A: Addison-Wesley, 1983, Vol.12, pp:127-168

[18] ZHOU Yue, SHEN Li, YANG Jie, Feature analysis m ethod of tongue im age by chinese medical diagnosis based on image processing, Infrared and Laser Enginering, 2002.12. Vol. 31 No.6 pp490-494

Distributed Computing Systems: P2P versus Grid Computing Alternatives

A. Cabani[*], S. Ramaswamy[+], M. Itmi[*], S. Al-Shukri[+], J.P. Pécuchet[*]

***Abstract*-Grid and P2P systems have become popular options for large-scale distributed computing, but their popularity has led to a number of varying definitions that are often conflicting. Taxonomies developed to aid the decision process are also quite limited in their applicability. While some researchers have argued that the two technologies are converging [1], in this paper, we develop a unified taxonomy along two necessary distributed computing dimensions and present a framework for identifying the right alternative between P2P and Grid Computing for the development of distributed computing applications.**

1. Introduction

A distributed computing system is defined as a collection of independent computers that appear to their users as a single computing system [2]. Distributed software systems are increasingly being used in modern day software systems development to tackle the issues of geographically separated work groups and increasing application complexities. Often this constitutes the interconnection of autonomous software residing on individual machines through communication networks to enable their users to cooperate and coordinate to successfully accomplish their objectives.

The widespread need for a distributed system based solution is due to the need for resource sharing and fault-tolerance. Resource sharing implies that a distributed system allows its resources - hardware, software and data – to be appropriately shared amongst its users. Fault-tolerance means that machines connected by networks can be viewed as redundant resources, a software system could be installed on multiple machines to withstand hardware faults or software failures [3].

For a distributed system to support active resource sharing and fault-tolerance within its multitude of nodes, it needs to possess certain key properties. These include openness and transparency. Openness in a distributed system is achieved by specifying its key interface elements and making it available to other software developers so that the system can be extended for use. Distributed systems generally tend to provide three forms of transparency. These include: (i) Location transparency, which allows local and remote information to be accessed in a unified way; (ii) Failure transparency, which enables the masking of failures automatically; and (iii) Replication transparency, which allows duplicating software/data on multiple machines invisibly.

To be able to demonstrate the above properties, in turn such a distributed system must provide support for concurrency and be built on a scalable architectural framework. Concurrency refers to the simultaneous processing of requests to multiple interconnected machines / networks. Scalability refers to the adoption of an interconnection network architecture that allows for seamless extendibility to a large number of machines and/or users to support the needs of increased processing power requirements.

In this paper, we compare and contrast two currently popular approaches for distributed computing applications: Grid and P2P approaches. The objective of both P2P and grid computing is the collective, coordinated use of a large number of resources scattered in a distributed environment. However the user communities that have adopted and popularized these two approaches are vastly different, both in terms of their user-level requirements as well as the architectural design of the systems themselves.

This paper is organized as follows: Section 2 and 3 present a brief overview of Grid and P2P computing. Section 4 presents the unified taxonomy along two necessary distributed computing dimensions. Section 5 compares and contrasts Grid and P2P computing using a set of commonly desired criteria for a distributed computing solution. Section 6 concludes the paper.

2. Grid Computing

According to IBM's definition [4]: "A grid is a collection of distributed computing resources available over a local or wide area network that appear to an end user or application as one large virtual computing system. The vision is to create virtual dynamic organizations through secure, coordinated resource-sharing among individuals, institutions, and resources. Grid computing is an approach to distributed computing that spans not only locations but also organizations, machine architectures and software boundaries to provide unlimited power, collaboration and information access to everyone connected to a grid."

Another definition, this one from The Globus Alliance is [5]: "The grid refers to an infrastructure that enables the integrated, collaborative use of high-end computers, networks, databases, and scientific instruments owned and managed by multiple organizations. Grid applications often involve large amounts of data and/or computing and often require secure resource sharing across organizational

* LITIS Laboratory EA4051, INSA-Rouen, 76131 Mt. St-Aignan Cedex, France. This work was supported, in part, by INSA-Rouen - for supporting Dr. Ramaswamy as a visiting professor at INSA during June 2006.
Email: adnane.cabani@insa-rouen.fr / itmi@insa-rouen.fr / jean-pierre.pecuchet@insa-rouen.fr

\+ Knowledge Enterprises for Scalable Resilient Infrastructures (KESRI), Computer Science Department, Univ. of Arkansas at Little Rock, AR 72204 USA. This work was supported, in part, by an NSF MRI Grant #CNS-0619069 and the Graduate Institute of Technology at UALR.
Email: srini@acm.org / shalshukri@ualr.edu

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 47–52.

boundaries, and are thus not easily handled by today's Internet and Web infrastructures."

Grid computing evolved out of the scientific realm where there was a need to process and analyze increasingly colossal quantities of data such as those needed to perform weather or climatic forecasts [6, 7], to model and calculate the aerodynamic behavior of a plane, in genomics [8], etc. Such application specific needs were not only at odds with the wait times for technological evolution but also were highly intolerant to the advances attained by means of Moore's law [9] with respect to performance (doubling every eighteen months), storage (doubling every twelve months), and networks (double every nine months). Since the speed of network performance outpaces that of processors and storage, it has led to the increasing interconnection of geographically scattered resources by means of a dynamic network that accumulates the capacities of calculation, storage, etc.

Grid computing [10-12] thus allows for better resource sharing between users in an institution to resolve a common problem. From a business perspective, the purpose of grid computing is to minimize the time to market, thereby profiting from the infrastructure costs incurred. Grid computing offers its users' access to processing across diverse storage structures which are transparently distributed geographically. It is based on the concept of on demand data processing wherein a user pays according to their needs and resource consumption.

3. Peer-to-peer Computing

As contrasted from grid computing, peer-to-peer computing refers to a network of equals that allows two or more individuals to spontaneously collaborate without necessarily needing any centralized coordination [13]. P2P computing was made famous by a music sharing frenzy generated by Napster, which was initially a server based centralized architecture. Other P2P systems have since appeared without the limitation imposed by a centralized server. Here, a user seeking a file (song, video, software) sends their query which is incrementally forwarded by the nodes of the network, thereby creating an ad-hoc chain between the requester's PC requester to the supplier's PC – culminating in the transfer of the requested file. Examples include Limewire [14], Kazaa [15], eDonkey [16], BitTorrent [17]; which are some popular P2P systems. When designing P2P applications, it is important to assume that peers are untrustworthy [18]. While they are designed to interconnect and communicate with each other they can join or quit dynamically from the P2P network. When a node quits, there will be communication failures. This makes the development of P2P applications a very challenging task.

While P2P technology can be applied to many application domains, most current utilization is customer targeted with the primary focus on file sharing. These systems allow files to be easily shared and quickly propagated through the Internet without powerful host servers. Other applications include:

1. Personal productivity applications: Collaboration between individual users, i.e. sharing address books, schedules, notes, chatting, etc. allows improvements in productivity. Connecting such desktop productivity software systems together enables collaborative e-business communities to form for flexible, productive, and efficient working teams. For example, Java developers have used OpenProjects.net to collaborate. On a broader scale, hundreds of thousands of uses use instant messaging, one of the most popular P2P applications to date.
2. Enterprise resource management: These systems allow the coordination of workflow processes within an organization thereby leveraging the existing network infrastructure for improvements in business productivity. For example, Groove [19] enables an aerospace manufacturers to post job order requests to partner companies and route the completed requests from one department to the next.
3. Distributed computation: A natural extension of the Internet's philosophy of robustness through decentralization is to design peer-to-peer systems that send computing tasks to millions of servers, each one possibly also being a desktop computer.

In [20, 21] the authors present a taxonomy for P2P applications and distinguish three specific categories of P2P applications. The specific classes include:

1. Parallel applications: In such applications, a large calculation is split into several small independent entities that can be executed independently on a large number of peers - SETI@Home [22], genome@Home, etc. Another possibility is making the calculation of the same operation but with different data sets or parameters. Such computation kind called a parametric study computation. Example: fluid dynamics simulation. The goal is to solve computational problems and cycle-sharing. P2P cycle-sharing and grid computing are converged but its origins are different [11, 23]. In P2P cycle-sharing the whole application runs on each peer and no need communication between the peers.
2. Content and file management: Content encapsulates several types of activities and refers to anything that can be digitized; for example, messages, files, binary software. It essentially consists of storing, sharing and finding various kinds of information on the network. The main application focus is content exchange. Such projects aim to establish a system of files for distribution within a community and

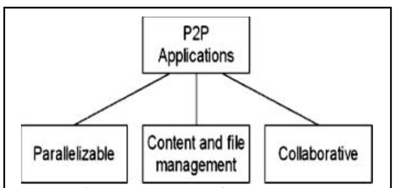


Fig. 1. Taxonomy of P2P Systems

examples include CAN [24] and Chord [25]. Other such applications are in distributed databases and distributed hash tables.

3. Collaborative: Collaborative P2P applications allow users to collaborate, in real time, without relying on a central server to collect and relay information. Such applications are characterized by ongoing interactions and exchanges between peers. Typical applications include: instant messaging (AOL, YM!). Some games (DOOM) have also adopted the P2P framework.

4. A Revised & Unified Taxonomy

There are several problems with the taxonomy presented in Fig. 1. While the taxonomy is very simplistic and coarse, there are three specific drawbacks. These include:

1. First, it lumps all parallel applications together into one classification. It is not in sufficient detail to help distinguish the application driven needs for distribution. Parallel applications may involve the distribution of the application itself over multiple nodes – as in cluster computing - to the locations where data is stored, or vice-versa – the distribution of the data elements to the applications reside – as in Grid computing. Computationally both of these types lead to completely different costs if not evaluated effectively. For the purposes of this paper, we concentrate on the distribution of the data elements.
2. A second issue is the need (or the lack of need) for synchronization. This issue is completely ignored in the present taxonomy. In fact, not all types of parallelization suit P2P applications. Whether the nodes are tightly coupled or loosely coupled is a very important criteria for the choice (or rejection) of a P2P solution. For tightly coupled applications, P2P is a bad implementation choice.
3. Another highly related issue that is ignored by this taxonomy is the issue of bandwidth disparity between member nodes. Certain studies have [26, 27] showed that P2P systems are extremely heterogeneous, but Grid computing systems tend to be more homogeneous in the composition of their nodes.

For these reasons, we propose a more well-refined and unified taxonomy as shown in Figure 2. This taxonomy is clearly defined along two dimensions. The first dimension relates to the requirements and capabilities of the underlying infrastructure, these include the architecture, the application domain and the high level of interaction required between the nodes. The second dimension relates to the various applicative constraints that are ignored by the existing taxonomy. While these criteria are independent by themselves, as seen from Figure 2, from the perspective of an application they are highly interdependent. These criteria include:

- Interconnectivity: P2P networks distinguish themselves by the presence of a volatile connectivity. Every peer can join or quit the system without any notice. Therefore, tightly or loosely coupled have an influence direct on expected results and techniques used for synchronization are not the same [28].
- Data size: The rate and size of data that is transferred between the nodes is an important discriminator for making an appropriate choice. Existing optimization techniques on P2P networks do not adequately address large scale keyword searches since the bandwidth required by such searches exceeds the internet's available capacity. Additionally, P2P-based systems thrive on low latency.
- Bandwidth: Available bandwidth is another critical criterion that needs to be considered.

This new taxonomy is helpful: the choice between grid computing and P2P is easier for developers. This requires three steps:

- After defining his application's objective, a developer chooses the application's kind (multicomputers, content and file management or collaborative). At this stage, the first dimension (requirements and capabilities) is fixed.

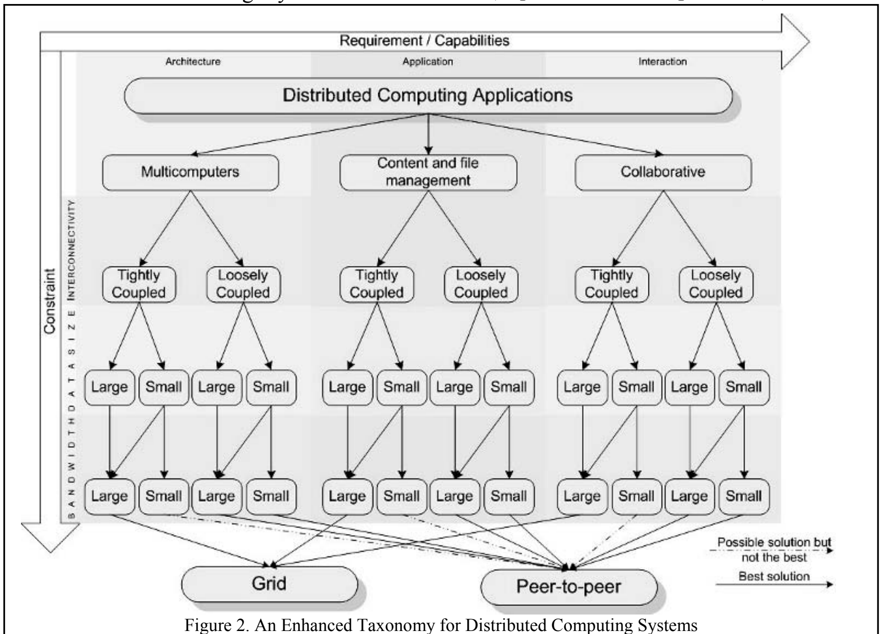


Figure 2. An Enhanced Taxonomy for Distributed Computing Systems

- Taking into account the three above criteria, he/she determines the application's specifications.
- Then, he/she chooses the adopted technology: grid computing or P2P.

5. Grid versus P2P Computing: A Feature-based Analysis

While it is clear that Grid and P2P computing are two promising approaches to distributed computing, they are very different and their differences are often misunderstood [29]. In a pure P2P system clients and servers work together and are indistinguishable from one another, this is not the case with Grid computing. One of the main P2P characteristics is that once the initial step is completed, data exchange is strictly and directly between peers. This property is completely absent in grid computing. Recently, P2P based Grid Computing systems that combine the advantages offered by P2P systems to scale up grid-based distributed computing systems have been proposed. Such systems enable the creation of a PC based grid architecture that address data overload problems caused by an enormous amount of access from clients by allowing for 'localized' data sharing by PCs through P2P communication mechanisms [30]. However, this may be largely dictated by several application domain characteristics. Hence in this section, we compare and contrast Grid and P2P computing using a set of commonly desired features of distributed computing solution. In [31] a similar attempt to identify and characterize the differences has also been reported – the characteristics include population, ownership discovery, user and resource management, resource allocation & scheduling, interoperability, single system image, scalability, capacity, throughput speed (lat. bandwidth). Figure 3 compares and contrasts P2P and Grid computing using several technical and economic characteristics. Some of the notable ones include the following:

1. *Decentralization:* Decentralization allows for flexibility and unlike client-server systems, does not suffer from single points of failure, wherein the server quickly becomes a bottleneck. Hence distributed systems that are scalable and resilient in unstable environments are very important. Decentralization allows us to move resources closer to where are accessed thereby decreasing response times and reducing, or even eliminating, network latency. This also allows better utilization of network capacity. Features that support decentralization include: Distribution of control, complete local autonomy and the ability orchestrate dynamic on the fly interactions.
2. *Cost and efficiency:* Performance-wise, networks are evolving at a faster rate that hardware and more PCs are connected to Web via broadband networks. This allows for better exploitation of these resources that were previously unrecognized. This has led to greatly increasing three important parameters in modern computing: storage, bandwidth and computing resources. This takes more sense if we know that it becomes easier to interconnect hundreds of million computers worldwide to form a network with the global revolution of the internet. Robert Metcalfe formulated an empirical law allowing measuring the utility of a network. Utility of a network = $k \times N^2$. In 1999, the Reed Law [32] adds a human dimension to the technological dimension. The utility of large networks, particularly social networks, can scale exponentially with the size of the network. The reason for this is that the number of possible sub-groups of network participants is $2^N - N - 1$, where N is the number of participants. All of these laws prove that number of connection peers is very important for increasing utility of a network. In P2P systems the number of nodes can reach hundreds of millions allowing increased network utility.
3. *Pervasive computing:* With P2P systems, it's possible to connect any machine with processor to network (PDA, cell phone, GPS...). That is there is a need for heterogeneous computing that is flexible enough to support new communication protocols for exchange of information in support of pervasive computing needs. Some work has been done on developing conceptual models for data management in pervasive computing environments based on cross-layer interaction between data management and communication layers [33].
4. *Target communities and incentives:* Although Grid technologies were initially developed to address the needs of scientific collaborations; commercial interest is growing [1]. Participants in contemporary Grids thus form part of established communities that are prepared to devote effort to the creation and operation of required infrastructure and within which exist some degree of trust, accountability, and opportunities for sanctions in response to inappropriate behavior. In contrast, P2P has been popularized by grassroots, mass-culture (music) file-sharing and highly parallel computing applications [14, 34] that scale in some instances to hundreds of thousands of nodes. The "communities" that underlie these applications comprise diverse and anonymous individuals with little incentive to act cooperatively.
5. *Resources:* In general, Grid systems integrate resources that are more powerful, more diverse, and better connected than the typical P2P resource [1]. A Grid resource might be a cluster, storage system, database, or scientific instrument of considerable value that is administered in an organized fashion according to some well-defined policy.
6. *Applications:* We see considerable variation in the range and scope of scientific Grid applications, depending on the interest and scale of the community in question [1].

7. *Keyword Searching:* Current P2P searching techniques in unstructured (examples: Gnutella and KaZaa [15, 35]) and structured P2P systems (examples: CAN [24], Chord [25], Pastry [36], Tapestry [37], and SkipNet [38]) include query flooding and inverted list intersection. In [39] the authors present a summary of techniques for unstructured P2P networks, while in [40] the authors present a search technique based on VSM (Vector Space Model) and LSI (Latent Semantic Indexing) for structured P2P systems. In [41], the authors identify storage and bandwidth as two limiting constraints on full text keyword searching on P2P systems. They suggest the use of a combination of optimizations and compromises to make this P2P searching feasible. Some hybrid schemes[42, 43] have been proposed for digital libraries but are not directly applicable to P2P systems. In [44] the authors propose a hybrid index multi-level partitioning scheme on top of structured P2P networks and indicate achieving a good tradeoff between partition-by-keyword and partition-by-document schemes.
8. *Scale, Security and failure:* Scalable autonomic management clearly has been achieved to a significant extent in P2P, albeit within specific narrow domains [1]. In [45] the authors analyze Gnutella's P2P topology graph and evaluate generated network traffic. They suggest that P2P systems must exploit particular distributions of query values and locality in user interests. They also suggest replacing traditional query flooding mechanisms with smarter and less expensive routing and/or group communication mechanisms. They report that Gnutella follows a multi-modal distribution, combining a power law and a quasi-constant distribution. Which makes the network as reliable as a pure power-law network when assuming random node failures, and makes it harder to attack by a malicious adversaries.
9. *Consistency Management:* Current P2P systems are based predominantly on sharing of static files. However, for using peer-to-peer networks in grid computing systems they will need to support sharing of files that are frequently modified by their users. Consistency has been studied for web caching [46, 47]. In [48] the authors present three techniques for consistency management in P2P systems: push (owner initiated) and pull (client initiated) and a hybrid push and adaptive pull technique.
10. *Services and infrastructure:* P2P systems have tended to focus on the integration of simple resources (individual computers) via protocols designed to provide specific vertically integrated functionality.

Feature / Criteria	**Grid Computing**	**P2P Computing**
GENERAL CRITERIA		
Goal	Virtual organization	Virtual system
Role of entities	Grid server	Peer both server and client
Number of entities	10 – 1000 users	Millions of users
Node	Dedicated	Not dedicated
Class	Pre-organized	Self-organized
TECHNICAL CRITERIA		
Structure	Static hierarchical	Fully distributed and dynamic
Fully decentralized	No	Yes
End-to-end connectivity	No	Yes
Scalability	Limited	Unlimited
Control mechanism	Central	Fully distributed
Connectivity	Static high speed	In/out anytime
Availability	High	Volatile
Failure risk	High	Low
Resources	More powerful	Less powerful
Resources discovery	Static central registration in a hierarchical fashion	Limited addition of a new peer on the network
Location transparency	Yes	No
Ad-hoc formation	No	Yes
ECONOMIC CRITERIA		
Communities	Established communities (closed)	Anonymous individuals (opened)
Participants	Registered	Voluntary
Reliability	Guaranteed trust	Partially (no trust peers)
Standards	Yes	No
Security	Secure	Insecure
Applications	Scientific – Data intensive	Compute cycles or file sharing

Fig. 3. A Comparative Criteria Driven Evaluation

6. Conclusions

In this paper our major contribution is an expanded taxonomy for the classification of distributed computing systems and highlighting commonalities and differences between Grid and P2P computing alternatives using a set of commonly desired features. Using these features, we have clearly identified and clarified issues to be addressed and the appropriate selection of the right alternative between P2P versus Grid Computing solution for the development of distributed computing applications.

7. References

[1] I. Foster and A. Iamnitchi, "On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing," *Lecture Notes in Computer Science,* vol. 2735, pp. 118-128, 2003.

[2] A. S. Tanenbaum, *Distributed Operating Systems*: Prentice Hall, 1994.

[3] G. Coulouris, J. Dollimore, and T. Kindberg, *Distributed Systems: Concepts and Design*: Addison-Wesley, 2001.

[4] L.-J. Zhang, J.-Y. Chung, and Q. Zhou, "Developing Grid computing applications, Part 1 : Introduction of a grid architecture and toolkit for building grid solutions," 2005.

[5] "Globus Alliance," http://www.globus.org.

[6] "Earth System Grid," http://www.earthsystemgrid.org/.

[7] "LEAD," https://portal.leadproject.org/gridsphere/gridsphere.

[8] "Genome@home, distributed computing," http://genomeathome.stanford.edu/.

[9] R. Hiremane, "From Moore's Law to Intel Innovation - Prediction to Reality," in *Technology@Intel Magazine*, 2005.

[10] M. D. Stefano, *Distributed Data Management for Grid Computing*: Wiley InterScience, 2005.

[11] A. Iamnitchi and I. Foster, "A Peer-to-Peer Approach to Resource Location in Grid Environments," in *Symp. on High Performance Distributed Computing*, 2002.

[12] I. Foster and C. Kesselman, *The Grid, Blueprint for a New Computing Infrastructure*: Morgan Kaufmann, 1998.
[13] D. Schoder and K. Fischbach, "Peer-to-peer prospects," *Commun. ACM,* vol. 46, pp. 27-29, 2003.
[14] "Limeware," http://www.limewire.com/.
[15] "Kazaa," http://www.kazaa.com/.
[16] "eDonkey," http://prdownloads.sourceforge.net/pdonkey/.
[17] "BitTorrent," http://www.bittorrent.com/protocol.html.
[18] M. Senior and R. Deters, "Market Structures in Peer Computation Sharing," in *Second International Conference on Peer-to-Peer Computing (P2P'02)*, Linköping, Sweden, 2002, pp. 128-135.
[19] "Groove," http://www.groove.net.
[20] D. S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu, "Peer-to-Peer Computing," HP, External HPL-2002-57, 2002.
[21] D. Barkai, "Technologies for sharing and collaborating on the Net," in *First International Conference on Peer-to-Peer Computing (P2P'01)*, 2001, pp. 13-28.
[22] "SETI@home," http://setiathome.ssl.berkeley.edu/.
[23] D. Talia and P. Trunfio, "Toward a synergy between P2P and grids," in *IEEE Internet Computing*. vol. 7, 2003, pp. 96, 94 - 95.
[24] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Adressable Network," in *ACM SIGCOMM 2001*, San Diego, California, United States, 2001, pp. 161-172.
[25] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications," in *ACM SIGCOMM 2001*, San Diego, California, United States, 2001, pp. 149-160.
[26] S. Saroiu, K. Gummadi, and S. D. Gribble, "A measurement study of peer-to-peer file sharing systems," in *Multimedia Computing and Networking*, San Jose, California, United States, 2002.
[27] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," *ACM SIGOPS Operating Systems Review,* vol. 35, pp. 131-145, 2001.
[28] V. K. Garg, *Concurrent and Distributed Computing in Java*: John Wiley & Sons, 2004.
[29] D. Barkai, *Peer-to-Peer Computing: Technologies for Sharing and Collaborating on the Net*: Intel Press, 2001.
[30] H. Sunaga, T. Oka, K. Ueda, and H. Matsumura, "P2P-Based Grid Architecture for Homology Searching," in *Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing (P2P'05) - Volume 00*, 2005, pp. 148-149.
[31] R. Buyya, "Convergence Characteristics for Clusters, Grids, and P2P networks," in *2nd Intnl Conf. on Peer-to-Peer Comp.* Linköping, Sweden, 2002.
[32] "ReedsLaw," http://www.reed.com/Papers/GFN/reedslaw.html.
[33] F. Perich, "On Peer-to-Peer Data Management in Pervasive Computing Environments," UMBC, 2004.
[34] D. Abramson, R. Sosic, J. Giddy, and B. Hall, "Nimrod: A Tool for Performing Parameterized Simulations Using Distributed Workstations," in *Fourth IEEE International Symposium on High Performance Distributed Computing (HPDC-4 '95)*, 1995, p. 112.
[35] "Gnutella," http://www.gnutella.com.
[36] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large scale peer-to-peer systems", IFIP/ACM Middleware. 2001. ." *Lecture Notes in Computer Science,* pp. 329-351, 2218.
[37] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area location and routing," UC Berkeley, Tech. Report UCB/CSD-01-1141 Apr. 2001.
[38] N. J. A. Harvey, M. B. Jones, S. Saroiu, M. Theimer, and A. Wolman, "SkipNet: A Scalable Overlay Network with Practical Locality Properties," in *4th USENIX Symp.on Internet Technologies and Systems (USITS '03)*, Seattle, WA, USA, 2003.
[39] B. Yang and H. Garcia-Molina, "Efficient Search in P2P Networks," in *22nd IEEE International Conference on Distributed Computing Systems (IEEE ICDCS'02)* Vienna, Austria: Computer Society, 2002.
[40] C. Tang, Z. Xu, and M. Mahalingam, "PeerSearch: Efficient information retrieval in peer-to-peer networks," in *Proceedings of HotNets-I, ACM SIGCOMM*, 2002.
[41] J. Li, B. T. Loo, J. Hellerstein, F. Kaashoek, D. R. Karger, and R. Morris, "On the Feasibility of P2P Web Indexing and Search," in *2nd Intl. Workshop on P2P Systems*, 2003.
[42] O. Sornil and E. A. Fox, "Hybrid partitioned inverted indices for large-scale digital libraries," in *Proc. of the 4th Int. Conf. of Asian Digital Libraries*, India, 2001.
[43] C. Badue, R. Baeza-Yates, B. Ribeiro-Neto, and N. Ziviani, "Distributed query processing using partitioned inverted files," in *9th String Proc. and Info. Ret. Symp. (SPIRE)*, 2002.
[44] S. Shi, G. Yang, D. Wang, J. Yu, S. Qu, and M. Chen, "Making Peer-to-Peer Keyword Searching Feasible Using Multi-level Partitioning," in *3rd Int. Workshop on Peer-to-Peer Systems*, San Diego, USA, 2004.
[45] M. Ripeanu, I. Foster, and A. Iamnitchi, "Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design," *IEEE Internet Computing Journal,* vol. 6, 2002.
[46] V. Duvvuri, P. Shenoy, and R. Tewari, "Adaptive Leases: A Strong Consistency Mechanism for the World Wide Web," in *Proc. of the IEEE Infocom'00*, Israel, 2000.
[47] J. Yin, L. Alvisi, M. Dahlin, and C. Lin, ", "Hierarchical Cache Consistency in a WAN," in *Proc. of the USENIX Symp. on Internet Technologies*, Boulder, CO, 1999.
[48] J. Lan, X. Liu, P. Shenoy, and K. Ramamritham, "Consistency Maintenance in Peer-to-Peer File Sharing Networks," in *Proc. of 3rd IEEE Workshop on Internet Apps*, 2002.

Context Sensitive Shape-Substitution in Nastaliq Writing System: Analysis and Formulation

Aamir Wali
University of Illinois at Urbana-Champaign
awali2@uiuc.edu

Sarmad Hussain
NUCES, Lahore
sarmad.hussain@nu.edu.pk

Abstract- **Urdu is a widely used language in South Asia and is spoken in more than 20 countries. In writing, Urdu is traditionally written in Nastaliq script. Though this script is defined by well-formed rules, passed down mainly through generations of calligraphers, than books etc, these rules have not been quantitatively examined and published in enough detail. The extreme context sensitive nature of Nastaliq is generally accepted by its writers without the need to actually explore this hypothesis. This paper aims to show both. It first performs a quantitative analysis of Nastaliq and then explains its contextual behavior. This behavior is captured in the form of a context sensitive grammar. This computational model could serve as a first step towards electronic Typography of Nastaliq.**

I. Introduction

Urdu is spoken by more than 60 million speakers in over 20 countries [1]. Urdu is derived from Arabic script. Arabic has many writing styles including Naskh, Sulus, Riqah and Deevani. Urdu however is written in Nastaliq script which is a mixture of Naskh and an old obsolete Taleeq styles. This is far more complex than the others.

Firstly, letters are written using a flat nib (traditionally using bamboo pens) and both trajectory of the pen and angle of the nib define a glyph representing a letter. Each letter has precise writing rules, relative to the length of the flat nib. Secondly, this cursive font is highly context sensitive. Shape of a letter depends on multiple neighboring characters. In addition it has a complex mark placement and justification mechanism. This paper examines the context sensitive behavior of this script and presents a context sensitive grammar explaining it.

A. Urdu Script

The Urdu abjad is a derivative of the Persian alphabet derived from Arabic script, which in itself is derived from the Aramaic script (Encarta 2000, Encyclopedia of Writing and [2]). Urdu has also retained its Persio-Arabic influence in the form of the writing style or typeface. Urdu is written in Nastaliq, a commonly used calligraphic style for Persio-Arabic scripts. Nastaliq is derived from two other styles of Arabic script 'Naskh' and 'Taleeq'. It was therefore named Naskh-Taleeq which gradually shortened to "Nastaliq".

Fig. 1. Urdu Abjad

II. Positional And Contextual Forms

Arabic is a cursive script in which successive letters join together. A letter can therefore have four forms depending on its location or position in a ligature. These are isolated, initial, medial and final forms. Consider the following table 1, in which letter 'bay' indicated in gray has a different shape when it occurs in a) initial, b) medial, c) final and d) isolated position. Since Urdu is an derived from Arabic script and Nastaliq is used for writing Urdu, both Urdu and Nastaliq inherit this property.

TABLE 1
Positional Forms For Letter *Bay*

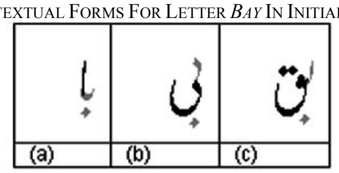
با	قبا	قب	ب
(a)	(b)	(c)	(d)

T. Sobh (ed.), *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, 53–58.

Letters ‘alif’, ‘dal’, ‘ray’ and ‘vao’ only have two forms. These letters cannot join from front with the next letter and therefore do not have an initial or medial forms.

Nastaliq is far more complex than the 4-shape phenomenon. In addition to position of character in a ligature, the character shape also depends on other characters of the ligature. Thus Nastaliq is inherently context sensitive. Table 2 below shows a sample of this behavior in which a letter bay, occurring in initial form in all cases, has three different shape indicated in grey. This context sensitivity of Nastaliq can be captured by substitution grammar. This is discussed in detail later in this paper.

TABLE 2
CONTEXTUAL FORMS FOR LETTER *BAY* IN INITIAL FORM



III. GROUPING OF ‘SIMILAR’ LETTERS

There are some letters in Arabic script and consequently in Urdu, that share a common base form. What they differ by is a diacritical mark placed below or above the base form. This can be seen in table 3 below which shows letters ‘bay’, ‘pay’, 'tay’, ‘Tey’ and ‘say’ in isolated forms. And it’s clearly evident that they all have the same base form. This is also true for initial, medial and final forms of these letters.

TABLE 3
LETTERS WITH SIMILAR BASE FORM

Isolated Form	ب	پ	ت	ٹ	ث
Initial form	بـ	پـ	تـ	ٹـ	ثـ
	(a)	(b)	(c)	(d)	(e)

Since these letters have a similar base shape, it would be redundant to examine the shape of all these letters in different positions and context. Studying the behavior of one letter would suffice the others. Table 4 below shows all groupings that are possible. The benefit of this grouping is that instead of examining about 35 letters in Urdu, only half of them need to be looked into. Note that only the characters that are used in place of multiple similar shapes are shown. The rest of the characters in the abjad are used without any such similar-shape classification.

TABLE 4
GROUPING OF LETTERS WITH SIMILAR BASE FORM

Similar Base Forms	Letter
ب پ ت ٹ ث Also ن and ی in initial and medial form	ب
ج چ ح خ	ج
د ڈ ذ	د
ر ڑ ز ژ	ر
س ش	س
ص ض	ص
ط ظ	ط
ع غ	ع
ک گ	ک

IV. METHODOLOGY: TABLETS

The Nastaliq alphabets for Urdu have been adapted from their Arabic counterparts as in the Naskh and T’aleeq styles from which it has been derived. However, even for Urdu, this style is still taught with its original alphabet set. When the pupil gains mastery of the ligatures of this alphabet, then he/she is introduced to the modifications for Urdu.

The methodology employed for this study is similar to how calligraphy is taught to freshmen. The students begin by writing isolated forms of letters. In doing so, they must develop the skill to write a perfect shape over and over again by maintaining the exact size, angle, position etc. When the students have achieved the proficiency in isolated form it is said that they have completed the first ‘taxti’, meaning tablet. The first tablet is shown in figure 2 below; ‘taxti’ or tablet can be considered as a degree of excellence. First tablet is considered level 0

Fig. 2. Tablet for Isolated forms

Once this level is completed, the students move over to level 2. In this level, they learn to write all possible two-letter ligatures. This phase is organized into 10 tablets. The first tablet consists of all ligatures beginning with ‘bay’, the second abjad of Arabic script. For example in English it would be like writing ba, bb, bc etc. all the way till bz. Note that the first letter in Arabic script is ‘alif’ which has no initial form and therefore does not form two letter ligatures beginning with it. See section 2 above. The bay tablet is shown in figure 3 below.

Fig. 3. Bay Tablet

In the similar way the students move over to the second tablet of this phase which has ligatures beginning with ‘jeem’ and so on. The ‘jeem’ tablet is also shown below in figure 4.

Fig. 4. Jeem Tablet

Note: If the students learn to write ligatures beginning with ‘bay’, then it is automatically assumed they can also write ligatures beginning with ‘tay’, ‘say’, etc because all these letters have the same shape as ‘bay’ and only differ in the diacritical mark. Please refer to section 3 above for details.

The level 0 and 1 helps to understand a lot about the initial and final forms of letters. There are however no further (formal) levels of any kind. So, in addition to understanding the medial forms, the students are also expected to learn to write three and above letter ligatures on their own through observation and consultation.

For this study, a calligrapher was consulted who provided the tablets and other text that was required.

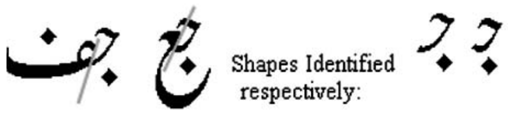
V. SEGMENTATION

Before moving over to the analysis of Contextual substitution of Nastaliq, one of the predecessor worth mentioning is segmentation. Since Nastaliq is a cursive script, segmentation plays an important role in determining the distinct shapes a letter can acquire. Consider a two letter ligature composed of jeem and fay. There can be a number of places from where this ligature can be segmented (as indicated by 1,2 and 3) giving different shapes for initial form of jeem.

(1) (2) (3)

If (1) is adopted as the segmentation scheme and being consistent with it, when this is applied on another two letter ligature that also begins with a ‘jeem’, it follows that both the ligatures share a common shape of ‘jeem’. Both the ligature are shown below with segmentation approach (1). The resulting, similar ‘jeem’ shapes is also given.

Where as if the second (2) option is accepted for segmentation, then the two initial forms will be different.



The segmentation approach selected for this analysis is the later one or approach (2). The reason for this selection is that the resulting shapes are a close approximation to the shapes in the calligrapher's mind. Most probably because these shapes represent complete stokes (, though an expert calligrapher may be able to write the whole ligature in one stroke). While in former case there seems to be some discontinuity in the smooth stroke of a calligrapher's pen.

The discussion on contextual shape analysis in next section is therefore based on the "stroke segmentation' approach.

VI. Contextual Analysis Of Nastaliq

This section lists the context sensitive grammar for characters occurring in initial position of ligatures.

Explanation of Grammatical Conventions:

The productions such as:

ب → ب$_1$ / __<A> | ف $_{\text{Initial Form}}$ __

is to be read as ب transforms to ب$_1$ (ب → ب$_1$), in the environment (/) when ب occurs before class A (__<A>) or (|) when ب occurs before initially occurring ف.

Note that 'OR' (|)operator has a higher precedence than 'Forward Slash' (/) operator. Thus, it would be possible to write multiple transformations in one environment using several 'OR' (|) operator on the right side of a single (/).

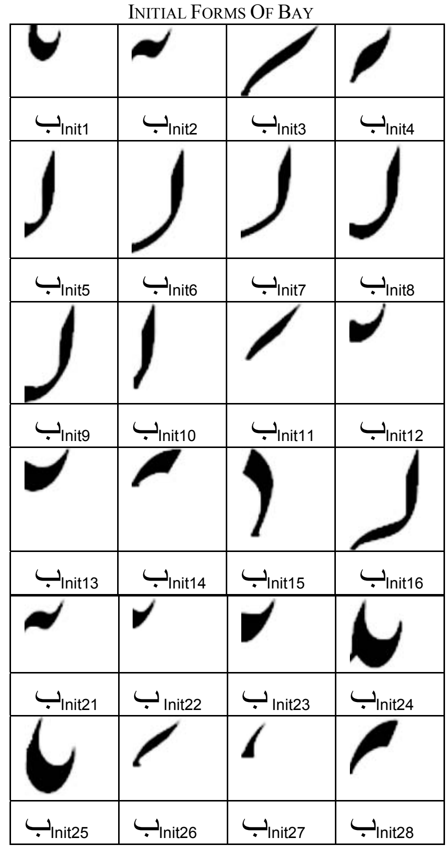
Contextual Shift

One invariant that have predominantly existed in this contextual analysis of Nastaliq is that *the shape of a character is mostly dependent on immediate proceding character*. That is given a ligature composed of character sequence $X_1, X_2 \ldots X_N$ for N>2, the shape of character X_i where i < N, is determined by letter X_{i+1}. While all preceding letters $X_1 \ldots X_{i-1}$ and character sequences after its following character i.e $X_{i+2} \ldots X_N$ have no (or little) role in its shaping. Sequence of bay's form an exception to this general rule. Other exceptions are also mentioned

Initial-Position characters:

The following table lists the initial shapes of letter 'Bay'. The last eight shapes were identified during analysis of three-character ligature and do not realize in two character ligatures. Another way of putting it is that these form can only occur in ligatures of length greater than or equal to three. There is no such restriction for the first 16 shapes.

TABLE 5
Initial Forms Of Bay



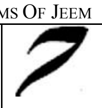
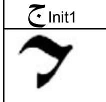
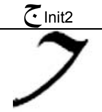
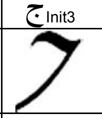
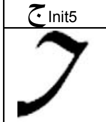
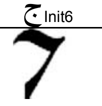
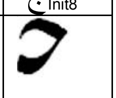
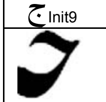
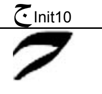
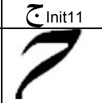
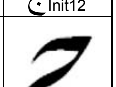
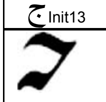
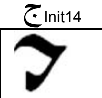
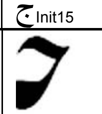
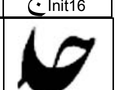
The context in which these shapes occur has been formulated in the form of a context sensitive grammar. The grammar for initial forms of 'bay' is given below. Note: The word 'medial' used in the grammar represents medial shapes of 'bay' given in appendix A

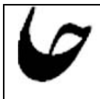
ب → ب $_{\text{Init1}}$ / __ ل __ / ک __ | د __ | ا
ب → ب $_{\text{Init2}}$ / __ ب$_{\text{Final}}$
ب → ب $_{\text{Init3}}$ / __ ج
ب → ب $_{\text{Init4}}$ / __ ر
ب → ب $_{\text{Init5}}$ / __ س
ب → ب $_{\text{Init6}}$ / __ ص
ب → ب $_{\text{Init7}}$ / __ ط

ب → ب Init8 / ___ ع
ب → ب Init9 / ___ ف
ب → ب Init10 / ___ ق | ___ و
ب → ب Init11 / ___ م
ب → ب Init12 / ___ ن
ب → ب Init13 / ___ ہ Final
ب → ب Init14 / ___ ھ
ب → ب Init15 / ___ ی
ب → ب Init16 / ___ ے
ب → ب Init22 / ___ ب Medial Only if none of the below apply
ب → ب Init21 / ___ ب Medial2
ب → ب Init23 / ___ ب Medial4
ب → ب Init24 / ___ ب Medial5
ب → ب Init25 / ___ ب Medial13
ب → ب Init26 / م د ___ | م ا ___ / م ل ___ / م ک
ب → ب Init27 / ___ م Otherwise
ب → ب Init28 / ___ ہ Medial

All other letters have similar number of shapes occurring in similar context. The initial forms of 'jeem' have been shown in table below. The grammar of 'jeem' can be derived from grammar of 'bay'. Likewise, shapes of other letters can be deduced from table 5 and table 6. Medial Shapes of Bay have been listed in Appendix A

TABLE 6
INITIAL FORMS OF JEEM

			
ج Init1	ج Init2	ج Init3	ج Init4
			
ج Init5	ج Init6	ج Init7	ج Init8
			
ج Init9	ج Init10	ج Init11	ج Init12
			
ج Init13	ج Init14	ج Init15	ج Init16
			
ج Init21	ج Init22	ج Init23	ج Init24

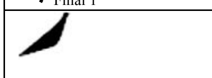
		Same as shape 11	
ج Init25	ج Init26	ج Init27	ج Init28

Note that the terminating shape of ب -init2 is very different from that of ج-init2. This is because they connect to different shapes of final 'bay'. This is discussed in the next section.

Final Forms:

With the exception of 'bay' and 'ray', all other letters have a unique final form. Both 'bay' and 'ray' have two final forms. These are shown in the table below. For 'bay', the first form occurs when it is preceded only by 'bay', (shape ب -init2 in the table above), 'fa', 'qaf', 'la' and 'ka', all in initial forms. While the other more frequent is realized else where and an example of letter connecting to it is 'jeem' having the form ج-init2. This explains the noticeable difference between the ending strokes of ب -init2 and ج-init2.

TABLE 7
FINAL FORMS FOR LETTER BAY AND RAY

	
ب Final 1	ب Final 2
	
ر Final 1	ر Final 2

The context for each of these shapes is given below. Interesting observation here is that 'ray-final1' occurs only when it is preceded by initial forms of 'bay' and 'jeem'. There is however no such precinct for letters 'ka' and 'la', which can be in any form (initial or medial).

ب → ب Final 1 / ب Initial Form ___
ب → ب Final 1 / ف Initial Form ___
ب → ب Final 1 / ق Initial Form ___
ب → ب Final 1 / ل Initial Form ___
ب → ب Final 1 / ک Initial Form ___
ب → ب Final 2 otherwise

ر → ر Final 1 / ب Initial Form ___
ر → ر Final 1 / ج Initial Form ___
ر → ر Final 1 / ک ___
ر → ر Final 1 / ل ___
ر → ر Final 2 otherwise

ACKNOWLEDGEMENT

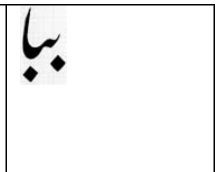
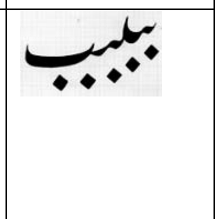
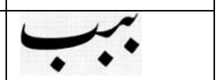
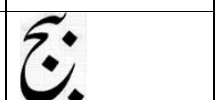
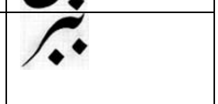
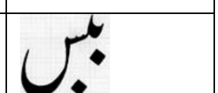
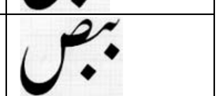
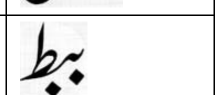
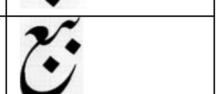
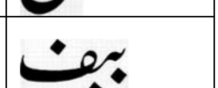
The work presented has been partially supported by a Small Grants Program by IDRC, APDIP UNDP and APNIC. The work on Nastaliq has been completed by the Nafees Nastaliq team at CRULP, NUCES.

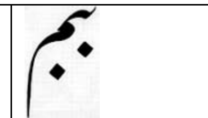
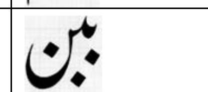
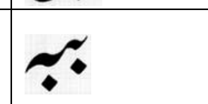
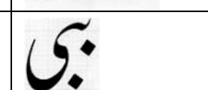
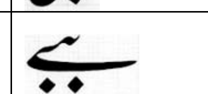
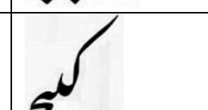
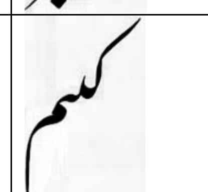
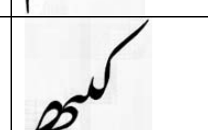
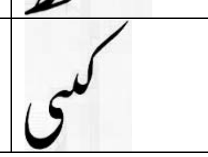
REFERENCE

[1] www.ethnologue.com
[2] http://en.wikipedia.org/wiki/Aramaic_script

APPENDIX A: MEDIAL SHAPES OF BAY WITH EXAMPLES

Given below is a detailed analysis of 20 medial shapes and 4 initial shapes of letter ‘bay’. These 4 initial shapes are mentioned here since they are used in medial position of a ligature also. Following table lists context sensitive grammar of a particular shape and the shape’s glyph it self when it occurs in medial position. Also included is an example corresponding to each glyph.

ب → ب Medial1 / __ <A> | __ ل | __ ب Init2 | __ ب Init6 | __ ب Init14 | __ ب Medial23 | __ ب Medial25	ب Medial1	
ب → ب Init2 / __ ب Medi1 | __ ب Medi2 | __ ب Medi8 | __ ب Medi9 | __ ب Medi10 | __ ب Medi12 | __ ب Init15 | __ ب Medi16 | __ ب Medi18	ب Init2	
ب → ب Medi2 / __ ب Final	ب Medi2	
ب → ب Medi3 / __ ج	ب Medi3	
ب → ب Medi5 / __ ر Final2	ب Medi5	
ب → ب Init6 / __ س	ب Init6	
ب → ب Medi8 / __ ص	ب Medi8	
ب → ب Medi9 / __ ط	ب Medi9	
ب → ب Medi10 / __ ع	ب Medi10	
ب → ب Medi11 / __ ف	ب Medi11	
ب → ب Medi12 / __ ق __ و	ب Medi12	

ب → ب Medi13 / __ م	ب Medi13	
ب → ب Init14 / __ ن Final	ب Init14	
ب → ب Init15 / __ ہ	ب Init15	
ب → ب Medi16 / __ ھ	ب Medi16	
ب → ب Medi17 / __ ی	ب Medi17	
ب → ب Medi18 / __ ے	ب Medi18	
ب → ب Medi23 / __ ب Medi3	ب Medi23	
ب → ب Medi25 / __ ب Medi5	ب Medi25	
ب → ب Medi33 / __ ب Medi13	ب Medi33	
ب → ب Medi36 / __ ب Medi16	ب Medi36	
ب → ب Medi37 / __ ب Medi17	ب Medi37	

Hyperbook authoring environment

Abdoulmajid Hakki
Lappeenranta University of Technology
Department of Information technology
ahakki@lut.fi

Abstract - **This paper will discuss the proposed XML-based authoring environment. An authoring environment is proposed to accommodate authors in creating structured content without familiarity of XML techniques**

1. INTRODUCTION

The world of publishing rapidly changing among the changes are the need of customised content or *Publishing on Demand*, the availability of different kinds of new publishing media.

In order to stay ahead of the changes in publishing industry the adaptation of database publishing techniques is needed. The content therefore needs to be structured, medium independent and reusable in order to make possible the automation of publishing. XML is suitable technology to achieve this [1].

Implementing this technology in the content authoring process in chorus needs to make sure that authors can deal with the changes brought by the implementation. With developing any technology we shall notice that the authors' main task is content creation. Must of them have little or no knowledge of XML. To require them create content using XML is quite unreasonable. The authoring environment must be simple and easy to use for authors to create structured content without being diverted or delayed from their main task.

An authoring environment is proposed to accommodate authors in creating structured content without familiarity of XML techniques.

This paper will discuss the proposed XML- based authoring environment. The paper is organised as follows: Chapter 2 discuss the act of writing and authoring need. Chapter three will describe the authoring models of Hyperbook. In chapter four the XML based Hyperbook authoring tool will be discussed. Chapter five is concentrated on conclusion and further work.

2. THE ACT OF WRITING AND AUTHORING NEED

Authoring for Hyperbook means creating the resources, labelling them combining them in a domain model or adaptive Hyperbook (AHB). An other step (probably by publisher) is to create a user model, responsible for customising the user. In order to help the creative work of authors and avoid technical requirements of authoring a book the authoring process has to be simple, efficient systematic, with clear semantic [2] and system independent.

AHB potentially allows writers to produce more *writerly* texts. Firstly HB offers the possibility of creating all kinds of multiple links between both the data assembled and the interpretative text which comment upon these data [3]. To assist book authors in creating content competently and effectively the general process of writing to be understand:

The author's target is to pass on information to readers. Each authors has a unique style and way of writing. Schriver [4] describes three traditions that influence writing:

The Craft Tradition (CT): This approach focuses mainly on writing techniques e.g guidelines, principles, styles, etc.). The CT emphasizes grammatical correctness and proper usage along with the ability to distinguish differences among styles of writing, various genres, modes of writing and techniques for exposition. In this tradition mastering these techniques and fundamentals are dominant.

The Rhetoric Tradition (RT): According to this tradition writing can be thought, learning to writes is a much perspiration as inspiration, and in order to write well, one need not to be born with a *gift.* The art of creation is fundamental. One has to acquire the (a) to create what to say about a subject, (b) to discover sensitive points of view for readers and (c) to develop clear arguments and descriptions.

The Romantic Tradition (RoT): Writing is successful when it expresses the inner vision of the writer in Romantic Tradition. Writing is seen as un-analyzable and un-teachable. Individuals either have the ability to write well or not. Writing is not something that you can just learn it. One must posses a unique, personal "gift" or genius to be able to write well. RoT basically focused on the idea of working by intuition.

Writers must identify content appropriate to the audience and to evaluate the quality of their drafts and route to a final product.

Understanding writing tradition can help in recognising the extraction of authors' hypothesis about writing.

The author's writing strategy is other fact to be considered in developing authoring tools. Chandler [5] surveys authors and categorizes their writing in to 5 strategis: The ***architectural strategy*** involves conscious pre-planning and organization, followed by writing out, with relatively limited revision. The ***bricklaying strategy*** involves polishing each sentence (or paragraph) before proceeding to the next. The completed text is not subjected to much subsequent revision. The ***oil painting strategy*** involves minimal pre-planning and major revision. Ideas are jotted down as they occur and are organized later. The ***water-colour strategy*** involves producing a single version of a text relatively rapidly with minimal revision. The ***Mixed strategy*** involves producing content of book using more than one strategy.

Understanding how authors write and what effect them will help in better accommodating their needs in an authoring environment. This understanding will help in developing an authoring environment, that would be better usable by authors.

3. AUTHORING MODELS OF HYPERBOOK

Today most authors are required to deliver their content digitally. In order to achieve these, authors requires assistance. Today word process has replaced typewriters. Authors are now used to work with them. This may make the transcoding [6] process to an authoring environment more difficult because authors will inevitably tend to generalise authoring tools.

The authoring environment aimed in this paper is not common word processor. Unlike Microsoft Word for example the environment which implements XML has a structure to which the created content must be integrated. This is one of most effective changes that cause the creative freedom of writers [7]. The author has three tasks: to carry out a creative process, to transfer of its results and to process received commends [7]. The creative process however usually lies totally within author's own domain. It arises

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 59–61.

during the brainstorming of authors. The problem is transferring author's idea into structured content.

Several solutions are offered for this problem. Michiel Schrage [8] classifies three kind of editors one can use: *syntax-directed* editors, *syntax-recognizing* editors and *hybrid* editors. Syntax-directed editors mainly support edit operations targeted at the document structure, whereas syntax-recognizing editors support edit operations on the presentation of the document. A hybrid editor combines syntax-directed with syntax-recognizing features. B. Usdin [9] highlights this point by justifying the need for semantics and not syntax. She proposes that an authoring tool avoids ambiguity, providing authors with as much functionalities as needed to create meaning. She suggests strictly defined rules and structures to hinder author's creative process. Callista [7] argues that Syntax-recognizing editors let users edit the presentation freely while the editor tries to recognise the document structure a paper. Because the document is derived from the presentation edit operation on the document are difficult to support.

Hybrid editor supports structural edit operations as well as presentation-oriented (often just textual) edit operations.

This research has attempts to study on implementation of a hybrid authoring environment. The hybrid editor takes a form freely structured environment simulating the blank page seen in Microsoft Word. Author can switch between editing the content and the structure showing the XML elements of work.

One of the critical parts of the authoring tool is the structure. In XML document structure is usually represented with DTD or an XML Schema [10]. These specify how the content is set in the document. Taking all the above factors into account one can come to the conclusion that there are two types of authoring environments: One whit a strict structured resembling a wizard [11, 12] and one with free structure. The wizard environment guides authors linearly during the development process. The author has therefore no choice but to create the specific form of content at a certain time [7]. While the freely structured environment let authors decide what he/she wants to create and when.

Figure 1 describes the writing process in a freely structured environment. As in figure 1 described the process of writing occur as follows:

1. The author creates a content, which is targeted to a certain target group according to the agreement between publisher and author.
2. The created frameworks are developed in chapters. The author also will specify some media specifications.
3. The metadata for chapters is created. The chapters are specified in to paragraphs.
4. During writing the author can develop content that was not specified in the frame work
5. After creating content the author or publishing editor may edit or restructure
6. After restructuring the content, the editor/author finalise the visualisation of the Hyperbook.
7. The content might be validated or the author adds a new content to the book.

The author shall have freedom to stop the process and deliver the final draft at any time during writing period of the content.

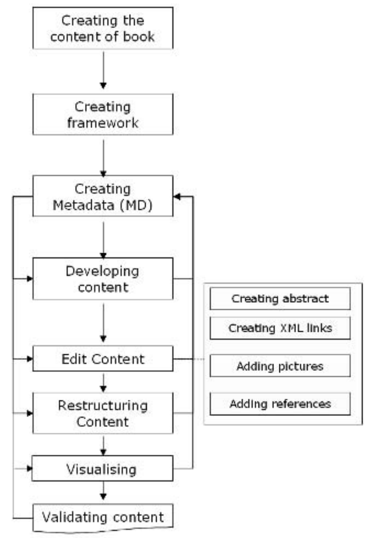


Fig. 1: A content creation model

4. XML BASED HYPERBOOK AUTHORING TOOL

Adaptive Hyperbook is not mature filed. However in adaptive hypermedia dozens of experimental and practical systems are developed. Toolkits, systems or shells that can be used by non-programming authors to develop an adaptive Hyperbook are too young. Before producing a real authoring tool the research have to be develop an explicit design approach that requires developing one or more Hyperbook. The framework is to be application independent which can be applied by their authors to rapid development of adaptive Hyperbooks in different domains.

The basic steps of creating an adaptive Hyperbook are not entirely different from steps in creating a hypertext system. In both cases the writer has to create the content object and to specify the links between them. As in previous section mentioned there are three major approaches used by authoring tools: Syntax-directed, syntax-recognizing and hybrid authoring tools.

Hierarchical structure of a book can be described an XML schema to take full advantages of XML processing tools, e.g. information in XML documents can be presented according to several different presentation styles and XML query languages provide facilities to retrieve data. An XML source document describing a Hyperbook presentation contains two kinds of specifications: The *Chapter Window, Image section.*

The Chapter Window (CW): The CW section contains the spatial layout of the book chapter in the presentation window (figure 2).

The Component Section (CS): The CS contains the description of the media objects involved in the chapter, their types, links to media files, etc. Media objects are organised in modules: each presentation contains at least one *module* enclosing continuous media objects, called images and pages [13].

Each element has a unique identifier *id*, which is used to reference the object from the other object sections of the document, a *type* and a *channel* in which it is viewed.

When XML specification describes a presentation, the images have an attribute *file* that value is the path of the book files. In figure 2, the image *XMLeditor* refers to media object which does not directly depend on chapter.

This page intentionally blank

```xml
<?xml version="1.0"?>

<Presentation xmlns="Hyperbook.xsd">
    <chapter>
    <title1>1. Introduction</title1>
    <title2>1.1. Background</title2>
    <paragraph>
        <key>XML</key> eXtensible Markup Language
    </paragraph>
    <paragraph>
        Web-maven
        <key>Elizabeth Castro</key>, who has penned Peachpit books on
        HTML,<key>Perl</key> and <key>CGI</key>, and
        Netscape, now tackles XML--an indispensable toolfor creating personalized,
        updated content for each visitor on yoursite. Whether you build
        Web pages for a living or you're taking on anew hobby, XML for
        the World Wide Web contains everything you need tocreate dynamic
        Web sites by writing XML code, developing custom <key>XMLapplications</key>
        with DTDs and schemas, transforming XML intopersonalized Web content through
        <key>XSLT-based</key>transformations, andprofessionally formatting XML
        documents with Cascading Style Sheets.The real power of XML lies in
        combining information from varioussources and generating personalized
        content for different visitors.Castro's easy-to-follow graphics show exactly
        what XML looks like,and her real-world examples explain how to transform and
        streamlineyour Web-site creation process by automatically updating content.
    </paragraph>
    </chapter>
<components>
    <module id="XML">
    <image id="XMLeditor file="XMLeditor.jpg" <channel="Image" type="pict"/>
    <composite id="section"
    <image>id="image" channel="image" type="pict"/>
    <page>id="caption" channel="caption" type="text"/>
    </module>
</components>
</Presentation>
```

Fig. 2 : XML schema for a book chapter

The objective of this research is to support authors on the creation of Hyperbook, which does not prescribe any instructional approach, variables or conditions for writers. As result developing a hybrid content editor that includes the definition of Hyperbook elements such as picture objectives, roles, key tags and so on. Moreover definitions of customizing properties and adaptive rules that will be considered to adjust the Hyperbook design shall be available.

6. CONCLUSION

Implementing and XML based authoring environment is not a simple task. Besides considering diverse writing styles, strategies and processes of the authors the demand of content is also to be considered. Therefore deciding the right environment that can technically support the writing process is a vital part of developing Hyperbook studies.

In this paper issues related to the query formulation and execution, XML schema and the integration are not discussed. These are of course problem of crucial importance and it is not claimed it is easy to formulate formally and to solve it.

In other hand any technical solution without authors' involvement can not fulfil its objectives. Therefore getting authors involve in the development process of the authoring tools is recommended. Understanding authors' perception may significantly reduce the risk of a failed research and implementation.

REFERENCES

1. Abdoulmajid Hakki, XML and the authoring environment of the Hyperbook, International Workshop on Applications of Semantic Web technologies for E-Learnin (2006)
2. 2. Alexandra Cristea, Authoring of Adaptive Hypermedia, Educational Technology & Society, 8 (3) (2005)
3. Bruce Mason & Bella Dicks, The digital Ethnographer, Cybersociology # 6 (2004)
4. Karen A. Schriver. Dynamics in Document Design: Creating Texts for Readers, John Wiley and Sons (1997)
5. Daniel Chandler, The Act of Writing, A Media Theory Approach, Prifysgol Cymru, Aberystwyth, University of Wales (1995)
6. Lev Manovich, The Language of New Media, Massachusetts Institute of Technology (2001)
7. Astrid Callista, Content Authoring in an XML-based and Author friendly Environment: Fact or Fiction? Institute of Information and Computing Sciences, Center for Content Knowledge and Engineering, Utrecht University, Netherlands (2004)
8. Martijn Michiel Schrage, Proxima, A presentation-oriented editor for structured documents, Institute for Programming research and Algorithmics (2004)
9. B. Tommie Usdin, When "It Doesn't Matter" means "It Matters", Extreme Markup Languages (2002)
10. Bradley, N. The XML Companion, Greath Briain, Addison-Wesley (2000)
11. D. Christopher Dryer , Wizards, Guides, and Beyond: Rational and Empirical Methods for Selecting Optimal Intelligent User Interface Agents, International Business Machines Corporation, Personal Software Products (1997)
12. Julien Stocq, Jean Vanderdonckt, WOLD: A Mixed-Initiative Wizard for Producing Multi-Platform User Interfaces, Proceeding of the 9th International Conference on Intelligent User Interface (2004)
13. Augusto Celentano and Ombretta Gaggi, A Synchronization Model for Hypermedia Documents Navigation, Dipartimento di Informatica, Universit& Ca' Foscari, Italy (2000)

XML and the architecture of the Hyperbook

Abdoulmajid Hakki
Lappeenranta University of Technology
Department of Information technology
ahakki@lut.fi

Abstract. **This paper presents architecture for development of Hyperbook, using XML to describe the application domain to adapt the content of Hyperbook to the user's behaviour. XML data-centric orientation makes it possible to describe application domain, data access and dynamic data composition functions. The system architecture of the Hyperbook is summarised. The characteristic of XML useful to manipulate data in a dynamic way is described. A general approach to data representation is described and the metadata and data presentation is presented.**

1 INTRODUCTION

Since the emergence of the World Wide Web, the concept of hypertext has become a main representation and presentation format for variety of applications. Hypertext books or Hyperbooks are among them, which are characterized as a grouping of electronic text which considered as entity [1]. In most cases, these hyperbooks still retain the conventional book structure and are partitioned into sub documents called chapters, sections, subsections or appendices. In [2] have been identified four properties of constructive hypermedia. These properties include the following:

1. **Intertextuality**: the process of interpreting one text by means of a previously composed text.
2. **De-centeredness and re-centeredness**: the points of focus depend on the interactive learners forcing the active learning processes
3. **Multivocality**: the networking, multi-perspective, and multi-media features of hypertext. This includes the multi-perspective, multi-channel, and crisscross capabilities.
4. **Malleability:** allowing learners to transform the presentation of information into personal representations of knowledge

The linking mechanism of Hyperbook offers users freedom so that it becomes necessary to offer support during navigation. To efficiently allow the realisation of user-adaptable contents and presentations, a clear separation between multimedia contents, domain model and user model should be achieved [3].

Basic Component of Hyperbook Systems are:

- The Adaptation domain Model [3]
- User Adaptation Model
- Bookmark Model
- Storage Model [4].

This paper presents architecture for development of Hyperbook, using XML to describe the application domain and interface model to adapt the content of Hyperbook to the user's behaviour. XML data-centric orientation makes it possible to describe application domain, data access and dynamic data composition functions.

The rest of paper is organised as follows: Section two summarises the system architecture of the Hyperbook. Section three describes the characteristics of XML useful to manipulate data in a dynamic way. Section four describes a general approach to data representation. Section five and six present the approach to metadata and adaptive presentation.

2 SYSTEM ARCHITECTURE

The main goal of Hyperbook system is to provide customised view of the content of the book responding to different preferences, interest and users. The application is to serve the strategy of active readying as the need of society in information age. In this respect a system to support Application Domain and User Adaptation Model is designed. The system has a three-tier architecture comprising the *Author Module, Application* and *Data layer* (Fig. 1). *Author Module* corresponds to the browser. It allows designing and validating the XML documents to create the content of Hyperbook (2.1).

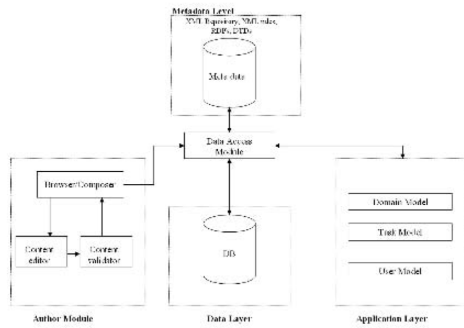


Fig. 1: Hyperbook system architecture - Application and Data Layers and Author Module.

The implementation of system is suggested either those concerning the on-line access to database, data composition and data delivery[1] or those allowing client-side elaboration (e.g. Java Applet).

2.1 AUTHORING MODULE

The Authoring Module (AM) comprise a content repository, which stores the content of Hyperbook files and an XML file which represent the structure of content.

The main object of Authoring Module is to allow design and validate (with respect to syntactic and semantic correctness the XML documents representing the concept of the Hyperbook. For simplifying the authoring process of Hyperbook the main components of Author Module are (fig. 1):

- The *Content editor (CE)*, which allows the content of Hyperbook.

[1]. E.g. JDBC, Java Servlet, Enterprise Java Beans, XML and SOAP .

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 63–66.

- The *Content validator*, which receives content description from CE and after validation of them, stores them in the Data layer (Data repository) [5].

2.2 APPLICATION MODULE

The task of Application layer is to provide integrated Hyperbook structure to the user. The application Layer consists of three main models:

- The *Domain Model* will be used to characterise the information accessible in the Hyperbook that is used to conclude in the user model.
- The *Task Model,* provide framework for structuring the content of Hyperbook and representing to user profile.
- The *User Model* stores individual users' information, preferences, goals and history.

2.3 DATA LAYER

The *Data Layer* (fig. 1) has two major means: the store of persistent data and to offer efficient access primitives. The content that stored to database consists of fine grained atomic units of the Hyperbook material. The aim of automatic units is to provide the maximum flexibility and reuse of these components in the variety of combinations [5].

2.4 METADATA LEVEL

The aim of using *Metadata* Level is to represent the logical structure of the data, as well as information concerning their usage and management (display, retrieval). A metadata is defined as a shared database of information about the content and data of the Hyperbook, that provided by the *Data Layer* or product by the Author. It stores: XML documents including the metadata, the Data Presentation Description, Schemes and XSL stylesheets [3, 5].

3 XML BINDING

Accessing Hyperbook from different devices will provide the basis for many important future in publishing industry. This will involve in some cases accessing local information that provide background about the local environment that are related to the content of Hyperbook. This service could be XML (extensible Markup Language) web service nature [6].

XML is a markup language for the construction of structured documents (fig. 2). It is a meta-language that allows us to create specialized markup languages for specific purposes [8]. XML is data-centric, which expresses the structure and eventually the meaning of the data contained in a document leaving its visualization to subsequent elaboration [9].

The Hyperbook application should be able to manage Modules from different types and domains of application using database technology which is suited to improve the usability and access to huge amounts of documents [10]. To make use of Hyperbook system, which architecture presented in previous sections, a format which allows storage, easy access, combination and adaptation of modules by a Hyperbook system is needed. The XML bindings are defined as a Document Type Declaration (DTD). The DTD definition was preferred to XML Schema [11].

The Hyperbook system should be able to manage *Adaptation Modules* from different types and domains of application using database technology which is particularly suited to improve especially the access to huge amounts of documents [12]. To make use of the advantages of our Hyperbook approach presented in the previous sections, we need a format which allows storage, easy access, combination and adaptation of *Modules* by a sophisticated system. XML document is a hierarchy comprising elements that have contents and attributes, XML is perfectly suited for representing the Hyperbook content, which is the conceptual content of the basic modules as well as the modular structure. The Hyperbook-specific model and its domain-specific instantiations serve as a well defined basis for a corresponding XML based markup language, the elements of which represent conceptual *Content Objects* by syntactical means. Furthermore the realization of View Specific by Style Sheets of the Extensible Style Language (XSL) [13] is used to Module Properties and Conceptual properties coming from XML bindings of different metadata proposals and standards as affirmed in [14].

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE ContentML SYSTEM "ContentML_simplified.dtd">
<ContentML>
      <ContentItem>
            <ContentItemId>uniqueId1234ABCD</ContentItemId>
            <Creator>tekija@domain.com</Creator>
            <DateTime>20060218</DateTime>
            <Language>fi</Language>
            <Category>Esimerkki sisällöt</Category>
            <Title>Asiasana merkkaus esimerkki</Title>
            <ContentData>
                  Tekstiä josta <Keyword>asiasana</Keyword> on merkattu.
                  <Keyword> Joukko erillisä asiasanoja
                        <KeywordAlias Value="asiasana"/>
                  </Keyword>
                  voidaan merkata siten että joukolle määritellään
                  yksi tai useampi alias.
            </ContentData>
      </ContentItem>
</ContentML>
```

Fig. 2. The XML content of the Hyperbook, created for mobile environment[2].

4 DATA REPRESENTATION

XML will play two distinct roles in Hyperbook development: The first role is that the infrastructure in which XML is used for data interchange and description. For example the content of book chapter are encoded in XML using SOAP and described in XML using WSDL (Web Services Description Language) [15]. The second role that XML plays could be named programming medium. In this programming role, the development of Hyperbook will be exposed to XML when designing the application interfaces in a top-down approach [15]. XML has several attractive features as a data representation and communication language, especially for Web applications.

The advantage of the XML is that it gives structure to unstructured data. It does this by embedding metadata tags in the data. Without these metadata tags, the fluid, unstructured contents of text documents would be difficult for Hyperbook application to

[2] The figure is part of Hyperbook project in Lappeenranta University of Technology for mobile environment, where six groups of students are participated. Figure 3. is one of examples, have been developed by group of students.

reference other than as a single body of text, or as a random collection of individual words. XML tags provide meaning to segments of data that are associated with a particular topic, usage or context and mark access points for data segment reference and retrieval [16]. Data stored as XML benefit from the numerous search techniques developed for XML formatted data including keyword searching, querying languages XPath and XQL, and linking functions XLink and XPointer [16].

Because XML is defined as a textual language rather than a data model, an XML documents always has implicit order—order that may or may not be relevant but is nonetheless unavoidable in a textual representation. A *well-formed*XML document places no restrictions on tags, attribute names, or nesting patterns [17].

5 CONTENT METADATA

Semantic annotations and metadata are seen as a crucial technique for transforming the World Wide Web from huge set of amorphous pages interlinked with each other to a semantic web, which uses semantic annotations in order to give meaning to these pages and their relationships [18]. This is in line with the development of Hyperbook, which semantic data models and schemata to define and give meaning to data and parts of programs.

The Hyperbook data related to the particular module are represented by the XML – based document according to its DTD [19]. Each element contains several conditions, which define the model of presentation according to the given context. Attributes and their values within each element represent conditions. The aim of Hyperbook is to experiment with attributes related to the type of data, granularity of presentation, time and spelling [19] of the presented information to several views.

6 ADAPTIVE PRESENTATION

The goal of Hyperbook study is to increase the functionality of Hyperbook by making it personalised. Adaptive Hyperbook will build a model of preferences of individual user and use this throughout the interaction of the content based on the needs of that user [4].

The critical feature of Hyperbook system is possibility of providing Hyperbook adaptation on the base of the user model [4].

The adaptive Hyperbook has to implement the following functionality:

- adaptive presentation
- adaptive navigation support
- adaptive information resource

The aim of the Adaptive presentation is to adapt the content of Hyperbook view to the user goals: device profile and other information stored in model [4]. In a system with adaptive presentation the views are not static, but adaptively generated or assembled from pieces for each user. The idea of adaptive navigation support is to guide the user in reading the Hyperbook by changing the appearance of visible link. Adaptive navigation as defined in can be considered as a generalisation of curriculum sequencing technology in Hyperbook context.

Adaptive information resource gives the authors and publisher editors appropriate information while developing the content of the book [4].

7 CONCLUSION

In this paper System Architecture to support the Application Domain and User Model was suggested to store XML based Hyperbook documents. The database schema is generic and independent of particular document type or schema. XML documents can be stored in a database where they are automatically clustered according to profile of users. The XML binding offer Hyperbook system is able to manage Adaptation Modules from different types and domains of application using database technology which is particularly suited to improve the access to huge amounts of documents.

The next work shall extend the current work by examining the details of the system architecture.

REFERENCES

1. Henze Nicola, Adaptive Hyperbooks: Adaptation for Project-Based Learning Resources, Universität Hannover 31 – 32, (2000)
2. Cecil Schmidt, Cognitive Flexibility Theory, Washburn University(2004)
3. Mario Cannataro, Andrea Pugliese, An XML-Based Architecture for Adaptive Hypermedia System Using a Probabilistic User Model, IEE 257 – 265, (2000)
4. Abdoulmajid Hakki, The importance of modeling metadata for Hyperbook, International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering CIS^{2}E 05, (2005)
5. Mario Cannataro, Alfredo Cuzzocrea, Andrea Pugliese, An probabilistic Approach to Model Adaptive Hypermedia System, ISI-CNR, Italy (2001)
6. Robert Steele, Yuri Ventsov, Tharam Dillon, XML Schema-based Discovery and Invocation of Mobile Services, IEEE (2004)
7. World Wide Web Constrium, Extensible Markup Language, Recommendation, www.w3.org, (2006)
8. Maria Elena Bonfigli, Giorgio Casadei, Paola Salomoni, Adaptive Intelligent Hypermedia using XML, 1500 ACM Symposium on Applied Computing, (March 10-21, 2000)
9. Mario Cannataro, Andrea Pugliese, An XML-Based Architecture for Adaptive Web Hypermedia Systems Using a Probablistic User Model, IEEE (2000)
10. Christian Süß: Adaptive Knowledge Management: A Meta-Modeling Approach and its Binding to XML. In: H.-J. Klein (Ed.), 12. GI-Workshop Grundlagen von Datenbanken, Plön, TR 1505, Christian-Albrechts-Universität Kiel, Germany (2000)
11. http://www.w3c.org/XML/Schema (2.2.2006)
12. S. Abiteboul, S. Cluet, V. Christophide, M. Tova, G. Moerkotte, and J. Siméon. Querying Documents in Object Databases. International Journal on Digital Libraries, 1(1):5–10 (1997)
13. C. Goldfarb and P. Prescod. *The XML Handbook*. Prentice Hall (1998)
14. LOM. IEEE P1484.12, Draft Standard for Learning Object Metadata (2002)
15. C. Lau, A. Ryman, Developing XML Web Services with WebSphere studio Application Developer, IBM Systems Journal, VOL 41, NO 2 (2002)

16. Garfinkel Yossi, Bridging the Data Divide, Leveraging XML to Build Integrated Aftermarket Parts Catalogs, Enigma, Inc., Israel (2001)
17. Roy Goldman, Jason McHugh, Jennifer Widom, From Semistructured Data to XML: Migrating the Lore Data Model and Query Language, Stanford University (1999)
18. Ora Lassila. Web metadata: A matter of semantics. *IEEE Internet Computing*, 2(4):30-37 (1998)
19. Mária Bieliková, Adaptive Presentation of Evolving Information Using XML, IEEE, 103 – 106 (2001)

A Novel Image Retrieval System Based On Dual Tree Complex Wavelet Transform and Support Vector Machines

Adeel Mumtaz, S.A.M. Gilani, Tahir Jameel
Faculty of Computer Science and Engineering
Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan
{adeel, asif, teejay}@giki.edu.pk

Abstract— **This paper presents a novel image retrieval system (SVMBIR) based on dual tree complex wavelet transform (CWT) and support vector machines (SVM). We have shown that how one can improve the performance of image retrieval systems by assuming two attributes. Firstly, images that user needs through query image are similar to a group of images with same conception. Secondly, there exists non-linear relationship between feature vectors of different images and can be exploited very efficiently with the use of support vector machines. At first level, for low level feature extraction we have used dual tree complex wavelet transform because recently it is proven to be one of the best for both texture and color based features. At second level to extract semantic concepts, we grouped images of typical classes with the use of one against all support vector machines. We have also shown how one can use a correlation based distance metric for comparison of SVM distance vectors. The experimental results on standard texture and color datasets show that the proposed approach has superior retrieval performance over the existing linear feature combining techniques.**

***Index Terms*—complex wavelet transform (CWT), support vector machine (SVM), content based image retrieval, texture image retrieval.**

I. Introduction

Digitization has made a profound effect in our everyday life from HDTV to digital camera. Global village concept is truly visualized in todays filed of communication where boundaries of many independent technologies have been merged to a common technology. With the rapid development of computing hardware, digital acquisition of information has become one popular method in recent years. Every day, G-bytes of images are generated by both military and civilian equipment. Large set of medical images, architectural and engineering designs, journalism and advertising, are worth mentioning. Consequently, how to make use of this huge amount of images effectively becomes a highly challenging problem [1]. Historically, only way to search through these collections was text based. Images were first annotated using text and then traditional Database Management Systems (DBMS) were used to retrieve relevant images when required.

There were two main problems with this approach; at first the amount of labor involved in manually annotating these images and secondly the inherent complexity and richness of image contents were making the annotation process difficult. For example it was not easy to label two images containing similar objects but with different meanings. In order to overcome these problems and to automate the retrieval process researchers from the field of computer vision proposed a new idea [2], which is known as content based image retrieval (CBIR). CBIR is also known as query by image content (QBIC) and content-based visual information retrieval (CBVIR). In these days many commercial and research CBIR systems are available. IBM's QBIC [7] and UC Berkeley's Blobworld are well known. Detailed comparison of such systems can be found at [3].

In content based image retrieval, we automatically extract features from images and then compare images using these features. Images having similar features would have similar contents. Basic block diagram of a content based image retrieval system is shown in Fig. 1.

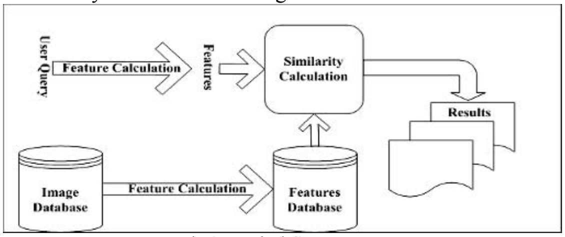


Fig.1. Typical CBIR system

At first features from all images in a database are extracted and stored into a feature database. This process is also known as indexing. When a user tries to search some images from the collection, he provides the system with a query image. Different options for query image are possible *i.e.*

- Image
- Rough Sketch
- Color or textural layout
- Verbal or semantic description

Features from query image are extracted by the same indexing mechanism. Then these query image features are matched with feature database using a similarity metric and finally similar images are retrieved.

A majority of indexing techniques are based on pixel

T. Sobh (ed.), *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, 67–72.

domain features such as color [4], texture [5], and shape [6]. However, with recent advancements in image compression, compressed domain indexing techniques gained popularity due to their less complexity. Some frequency domain techniques include wavelet domain features, Gabor transform and Fourier domain features. Comprehensive survey of existing CBIR techniques can be found in [1, 2].

Researchers have shown that texture is one of the most important features for CBIR. Texture refers to the visual patterns that have properties of homogeneity that do not result from presence of only single color or intensity. It is an innate property of virtually all surfaces, including clouds, trees, bricks, hairs, fabric, etc. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment. There are many review papers exist on texture based image retrieval. Manjunath and Ma [8] evaluated the texture image annotation by various wavelet transform representations and found that Gabor transform was the best among the tested candidates.

Kingsbury [13] proposed a new complex wavelet transform which gives a fast way of computing Gabor-like wavelets. Peter and Kingsbury [14] in his paper have shown that how one can use this new transform to speedup and enhance the image retrieval process. Kokare et al. [15] have proposed even better extension of this work. Janney et al. [17] have shown how we can enhance the texture extraction capabilities of CWT for color image retrieval. They have shown that we can achieve almost the same precession for color image retrieval as well. These properties of CWT have motivated us to use it as feature extraction for our proposed system.

Mostly during the comparison phase of features, linear metrics like Euclidean distance etc. were used. Recently Han et al. [9] has shown how one can improve the performance of image retrieval systems by assuming non-linear relationship among feature vectors and grouping the images into similar classes. We have applied a similar idea for retrieving of texture images. We have used support vector machines for classification of images in the database.

The paper is organized as follows: Section II provides a brief introduction to dual tree complex wavelet transform and some of its applications. Section III provides an overview of support vector machines and how SVM can be used for classification. Section IV explains feature extraction process. Section V describes proposed and implemented CBIR system. Section VI discusses the results of our technique in comparison with existing techniques. Section VII gives the concluding remarks.

II. Dual tree complex wavelet transform

Kingsbury's [13] dual-tree complex wavelet transform (CWT) is an enhancement to the discrete wavelet transform (DWT), with important additional properties. The main advantages as compared to the DWT are that the complex wavelets are approximately shift invariant (meaning that our texture features are likely to be more robust to translations in the image) and that the complex wavelets have separate sub-bands for positive and negative orientations. Conventional separable real wavelets only have sub-bands for three different orientations at each level, and cannot distinguish between lines at 45º and -45º.

The complex wavelet transform attains these properties by replacing the tree structure of the conventional wavelet transform with a dual tree. At each scale one tree produces the real part of the complex wavelet coefficients, while the other produces the imaginary parts. A complex-valued wavelet $\psi(t)$ can be obtained as:

$$\psi(t) = \psi_h(t) + j\psi_g(t) \tag{1}$$

where $\psi_h(t)$ and $\psi_g(t)$ are both real valued wavelets.

CWT like Gabor transform have six orientations at each of four scales (any number of scales can be used, but the number of orientations is built into the method). The main advantage as compared to the Gabor transform is speed of computation. It has a redundancy of only 4 in 2-dimensions and so the post-processing stages (of calculating mean and standard deviation) are also faster as it has less redundancy than the Gabor wavelets. Fig. 2 shows magnitudes of CWT coefficients for a texture image, one can see more details about orientation and scales. Each row represents one scale and columns represent angles within that scale.

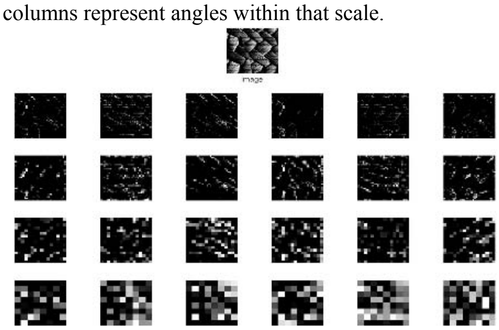


Fig.2. Four-scale CWT of a texture image

III. Support vector machines

There exists many pattern matching and machine learning tools and techniques for clustering and classification of linearly separable and non-separable data. Support vector machine (SVM) is a relatively new classifier and is based on strong foundations from broad area of statistical learning theory [11]. Since its inception in early 90's, it is being used in many application areas such as character recognition, image classification, bioinformatics, face detection, financial time series prediction etc.

SVM offers many advantages as compared with other classification methods such as neural networks. Kashif and Nasir [11] highlights many of advantages of support vector machines, these includes:

- Computationally very efficient as compared with other classifiers especially neural nets**.**
- Work well even with high dimensional data, a factor which limits many efficient classifiers.
- Can work well with less number of training data.
- Attempts to minimize test error rather than training error.
- Very robust against noisy data (noise greatly degrade the performance of neural nets).
- Curse of dimensionality and over fitting problems does not occur during classification.

Fundamentally SVM is a binary classifier but can be extended for multi-class problems as well. The task of binary classification can be represented as having, (Xi, Yi) pairs of data. Where $Xi \in X^p$, a p dimensional input space and $Yi \in [-1, 1]$ for both the output classes. SVM finds the linear classification function $g(x)=W.X+b$, which corresponds to a separating hyperplane $W.X+b=0$, where W and b are slope and intersection. SVM unlike other classifiers finds the optimal hyperplane, examples of optimal and non-optimal hyperplanes is shown in Fig. 3.

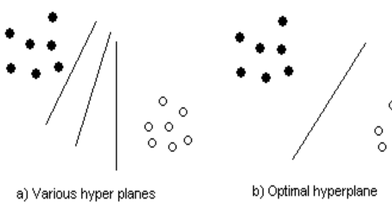


Fig.3. Optimal and non-optimal hyper-planes

SVM usually incorporate kernel functions for mapping of non-linearly separable input space to a higher dimension linearly separable space. Many kernel functions exist such as radial bases functions (RBF), Gaussian, linear, sigmoid etc. Different options exist to extend SVM for multi-class cases, these includes one against all, one against one and all at once. Fig. 4 shows how one against all SVM can be used for grouping of different classes inside an image database. Each support vector machine separates one class of images from rest of the database, which is shown by non-linear boundaries.

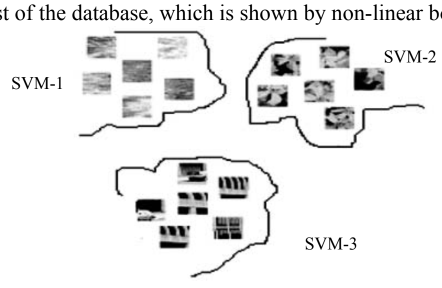


Fig. 4. One against all classification

IV. IMAGE CONTENT REPRESENTATION

First we performed a four scale (six angles) CWT on an image. We get 24 real and 24 imaginary detailed sub-bands, and 2 real and 2 imaginary approximation sub-bands. By taking the magnitudes of corresponding real and imaginary coefficients of both approximation and detailed sub-bands we get 26 sub-bands. To calculate the features we measure the mean and standard deviation of the magnitude of the transform coefficients in each of 26 sub-bands, in the same way as [14]. For color images we applied above process on each RGB color channels to get the feature vector.

V. CWT AND SVM BASED IMAGE RETRIEVAL SYSTEM

In this section, we describe the structure of the proposed SVMBIR system in detail. Fig. 5 shows the main components of the proposed system and the control flows among them.

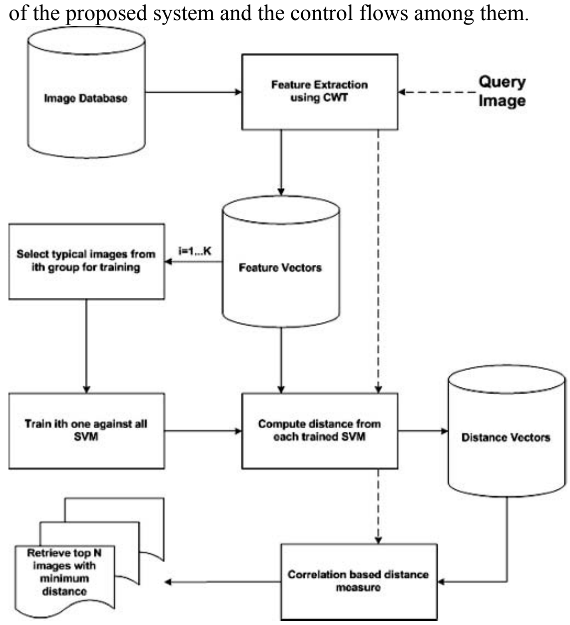


Fig. 5. The structure of the proposed SVMBIR system

The proposed system is based on SVM classifier. Our system is based on similar assumption as Han et al.[9] that is images users need are often similar to a set of images with the same conception instead of one query image and the assumption that there is a nonlinear relationship between different features. Following steps shows the detail of our proposed algorithm:

Step 1: Features are extracted from each image present in the image database using the aforementioned feature extraction process. These features are then stored in feature database for later comparison.

Step 2: From each class of images present in the image database some typical images (K) are selected for training of support vector machine for that class. We used one against all training method as it is the best when one needs good speed and reliable performance. This is done using *trainlssvm* function of LSSVM [16]. We used 'RBF' as kernel function for training of support vector machines. Optimal parameter selection is always a bottleneck of support vector machines. LSSVM provides a function *tunelssvm* which can be used for estimation of optimal parameters. We used grid search approach for searching of optimal parameters. Our used and final parameter values are 100 and 20 for *gam* and *sig2* respectively.

Step 3: In this step distance of each image present in the database from each SVM is calculated. This is done using *simlssvm* function of LSSVM. Each of this distance is grouped in the form of distance vectors. Finally we store all these distance vectors in distance vectors database.

Steps 1-3 are done offline and after these steps our system is ready to process the user queries.

Step 4: When the user give the system a query image, features from the query image are extracted. Using this feature vector of query image distance vector of query image is calculated.

Step 5: Query image distance vector is compared with all the distance vectors present in the distance vector database using correlation based similarity metric. Experimentally it is found that the following correlation metric performs the best for this comparison.

$$d_{rs} = 1 - \frac{(x_r - \bar{x}_r)(x_s - \bar{x}_s)'}{\sqrt{\left[(x_r - \bar{x}_r)(x_r - \bar{x}_r)'\right]\left[(x_s - \bar{x}_s)(x_s - \bar{x}_s)'\right]}} \quad (2)$$

This equation is one minus the correlation coefficient between vectors x_r and x_s. In our case x_r is the query image distance vector and x_s is the distance vector of images present in the database. *S* varies from 1…*N*, where *N* is the total number of images present in the image database. $\bar{x}_r$ and $\bar{x}_s$ are the means of the vectors x_r and x_s.

Step 6: Finally top *Q* images having minimum distance are retrieved and presented to the user.

VI. Experimental Results

In this section we have shown some experimental results to evaluate the performance of our proposed system.

For texture images we have used the same dataset as was used by Peter and Kingsbury [14]. The texture data set used in the experiments contains 100 texture images from the Massachusetts Institute of Technology (MIT) VisTex [12] database. Each 512x512 image is divided into 16 smaller images of size 128x128 giving a total of 1600 texture images in the database. Each original image is treated as a single class and so we have 16 examples from each of our 100 classes. For color images we used the same dataset as was used by Janney et al. [17]. The color image data set used in the experiments contains 7200 color images from the Columbia University Image Library (COIL-100) [18]. These images are organized as having 100 objects photographed at 72 different angles giving total of 7200 images.

For performance comparison we have used the same technique as was used by Peter and Kingsbury [14]. They used plots similar to precision recall graphs. For each of the 1600 images in the texture database we compute the distance to all of the other images and select the *N* nearest neighbors for each image. We then count how many of these belong to the correct class (up to a maximum of 15) and define the retrieval rate as this number divided by 15. This gives us a retrieval rate for each of the 1600 images and we simply average these rates to give an overall retrieval rate. In our results we plot the retrieval rate for a number of choices of *N*. A good set of features will give a retrieval rate that rapidly rises to 1. For each object present in coil-100 database we selected first instance out of 72 instances giving total of 100 query images and rest of the process is same.

Fig. 6 shows the comparison of our technique with Peter and Kingsbury's technique. In this comparison we used only 25% images from each class for training of support vector machines. As in Fig. 6 our technique performs better. We believe that the results can be significantly improved if we use more images for training. Fig. 7 shows the comparison with 50% training images.

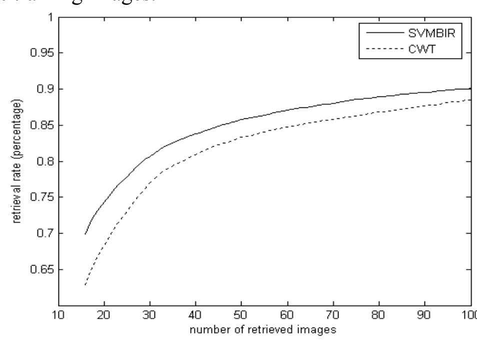


Fig. 6. Retrieval rate comparison using 25% training images

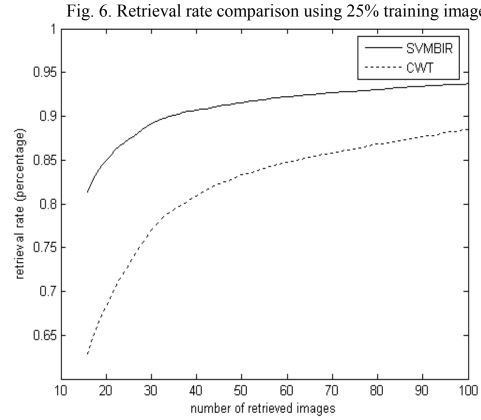


Fig. 7. Retrieval rate comparison using 50% training images

Table 1 shows error rate (number of mismatched images) comparison of our technique (with 25% training images) and Peter and Kingsbury's technique, for different types of texture images. We used 100 query images one from each class. Results are accumulated for different categories of images for example Bricks include 'Brick.0002', 'Brick.0003', 'Brick.0004', 'Brick.0005', 'Brick.0006', and 'Brick.0007' images of the VisTex database. It is very much clear from the table that our technique performed well almost in each category.

TABLE 1
RETRIEVAL PERFORMANCE FOR DIFFERENT CATEGORIES OF IMAGES

IMAGE CATEGORY	% ERROR SVMBIR	% ERROR CWT [14]
Bark	12.5	21.87
Brick	15.62	44.79
Buildings	39.84	58.59
Clouds	28.12	50
Fabric	5.46	28.12
Flowers	18.75	33.03
Food	20.62	27.5
Grass	41.66	54.16
Leaves	28.36	39.90
Metal	15.62	33.33
Misc	6.25	43.75
Paintings	22.91	29.16
Sand	11.60	17.85
Stone	6.25	6.25
Terrain	57.81	68.75
Tile	43.75	53.12
Water	13.75	37.5
WheresWaldo	70.83	70.83
Wood	0	0
Average	24.19	37.81

Fig. 8 shows the comparison of our technique with Janney et al. [17]. In this comparison we used only 12.5% images from each class for training of support vector machines. As in Fig. 8 one can see significance improvement in results.

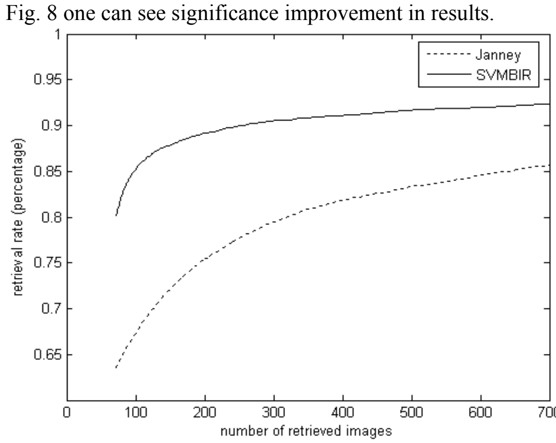


Fig. 8. Retrieval rate comparison using 12.5% training images

Here we also believe that the results can be significantly improved if we use more images for training. Fig. 9 shows the comparison with 25% training images. This shows we can almost get over 90% precision by using the idea presented in this paper.

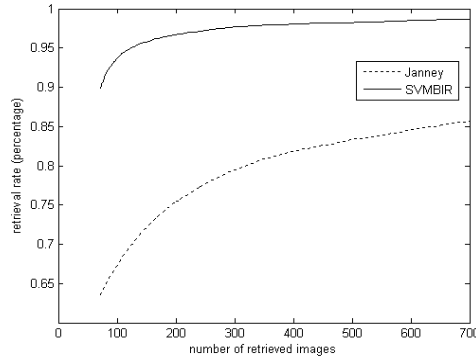


Fig. 9. Retrieval rate comparison using 25% training images

Table 2 shows error rate (number of mismatched images) comparison of our technique (with 12.5% and 25% training images) and Janney's technique, for different types of color images. We used 20 query images from different classes. Names of query image file in coil-100 database are also shown in the table. It is very much clear from the table that our technique performed well almost in each category.

TABLE 2
RETRIEVAL PERFORMANCE FOR DIFFERENT CATEGORIES OF IMAGES

IMAGE FILE NAME	% ERROR JANNEY [17]	% ERROR SVMBIR 12.5% TRAINING IMAGES	% ERROR SVMBIR 25% TRAINING IMAGES
Obj5__0.png	6.9444	4.1667	0
Obj10__0.png	22.222	0	0
obj15__0.png	45.833	22.222	13.889
Obj20__0.png	40.278	19.444	13.889
Obj25__0.png	0	0	0
Obj30__0.png	0	0	0
Obj35__0.png	0	0	0
Obj40__0.png	48.611	16.667	13.889
Obj45__0.png	76.389	69.444	33.333
Obj50__0.png	0	0	0
Obj55__0.png	63.889	30.556	15.278
Obj60__0.png	77.778	61.111	38.889
Obj65__0.png	75	43.056	31.944
Obj70__0.png	0	0	0
Obj75__0.png	30.556	0	0
Obj80__0.png	56.944	58.333	44.444
Obj85__0.png	38.889	30.556	16.667
Obj90__0.png	65.278	25	11.111
Obj95__0.png	0	0	0
Obj100__0.png	40.278	33.333	25
Average	**34.444**	**20.694**	**12.917**

Fig. 10 and Fig. 11 show the retrieved results of the proposed SVMBIR system, in which the first image is the query image. In Fig. 10, the query image is a texture image, while in Fig. 11 the query is a color image. We can see from these figures that the proposed system is very efficient as set of images with same conception can be retrieved, which is more similar to the human visual system.

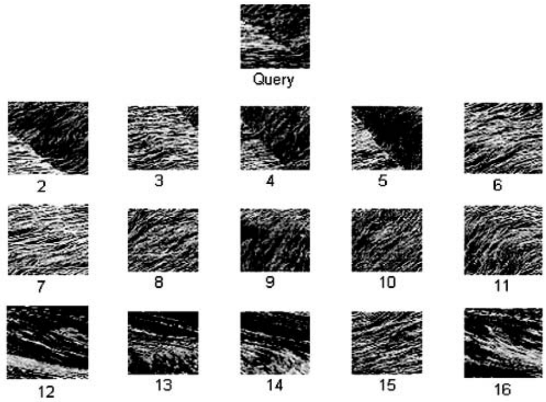


Fig. 10. Retrieved results of SVMBIR system for "*Query"* image

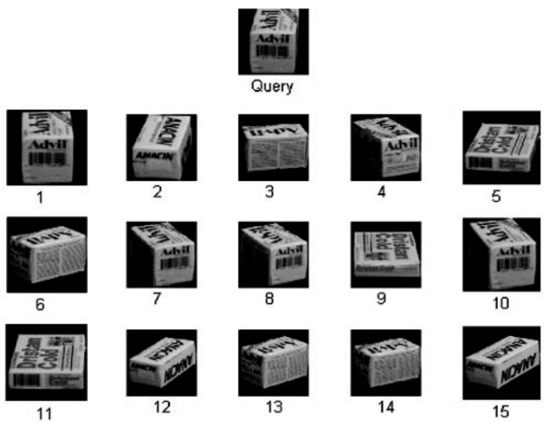


Fig. 11. Retrieved results of SVMBIR system for "*Query"* image

VII. CONCLUSION

In this paper, we presented a novel dual tree complex wavelet transform and support vector machine based image retrieval system. Proposed system is based on the observation that the images users need are often similar to a set of images with the same conception instead of one query image and the assumption that there is a nonlinear relationship between different features. In addition, we have shown how a correlation based distance measure can be used to enhance the retrieval accuracy. Finally, we compare the performance of the proposed system with other image retrieval system. Experimental results show that it is more effective and efficient to retrieve visually similar images having non-linear relationship among their feature vectors.

ACKNOWLEDGMENT

We thank Ghulam Ishaq Khan Institute (GIKI) for providing very conducive research environment as well as financial and moral support.

REFERENCES

[1] Y. Rui and T. S. Huang, "*Image Retrieval: Current Techniques, Promising Directions, and Open Issues*", *Journal of Visual Communication and Image Representation*, Vol.10, pp.39-62, 1999

[2] S. Deb, Y. Zhang, "An Overview of Content-based Image Retrieval Techniques," aina, vol. 01, no. 1, 2004.

[3] R.C. Veltkamp, M. Tanase, "*Content-Based Image Retrieval Systems: A Survey*", UU-CS-2000-34, Department of Computer Science, Utretch University. October 2002.

[4] A. Del Bimbo, M. Mugnaini, P. Pala, F. Turco, L. Verzucoli, "Image Retrieval by Color Regions," ICIAP 97, Image Analysis and Processing. 9th International Conference, ember, Florence, Italy.

[5] Georgy L. Gimel'farb, Anil L. Jain, "On retriving textured images from an image database," Patter Recognition, Volume 29 Number 9, pp. 1461-1483, september 1996.

[6] Venkat N. Gudivada, Vijay V. Raghavan, "Content-based Retrieval Systems," Computer, pp. 24-33, september 1995.

[7] Flickner M., Sawhney H., et al. (1995) "*Query by Image and Video Content: The QBIC System*", IEEE Computer, September.

[8] B. S. Manjunath and W. Y. Ma, "*Texture features for browsing and retrieval of image data*", IEEE Trans. Patt. Anal.Mach. Int. Special Issue on Digital Libraries, vol. 18, No. 8, pp. 837-842, August 1996.

[9] J.-H. Han, D.-S. Huang, T.-M. Lok, and M. R. Lyu, "*A Novel Image Retrieval System Based on BP Neural Network*" International Joint Conference on Neural Networks (IJCNN 2005), Montréal, Québec, Canada, July 31-August 4, 2005.

[10] Pranam Janney, G. Sridhar, V. Sridhar., "*Enhancing capabilities of Texture Extraction for Color Image Retrieval*". WEC (5) 2005: 282-285

[11] Kashif M. Rajpoot, Nasir M. Rajpoot, "*Wavelets and Support Vector Machines for Texture Classification*" in Proceedings of 8th IEEE International Multi-topic Conference (INMIC'2004)

[12] MIT-VisTex-database, http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html

[13] Kingsbury, N.G.,"*The Dual Tree Complex Wavelet Transform: a new efficient tool for image restoration and enhancement*",Proc. European Signal Processing Conf., pp319-322,Spetempber 1998.

[14] Peter, R. and Kingsbury, N.,"*Complex Wavelets Features for Fast Texture Image retrieval*", Proc IEEE Int. Conf. on Image Processing, Oct 25-28, 1999.

[15] Kokare, M., Biswas, P.K., Chatterji,B.N., "*Texture Image Retrieval Using New Rotated Complex Wavelet Filters*", SMC-B(35), No. 6, December 2005, pp. 1168-1178.

[16] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, Least Squares Support Vector Machines, World Scientific, Singapore, 2002 (ISBN 981-238-151-1)

[17] Pranam Janney, G. Sridhar, V. Sridhar, "Enhancing Capabilities of Texture Extraction for Color Image Retrieval" in Proceedings of World Enformatika Conference (Turkey) April, 2005.

[18] Columbia object image library (COIL-100), http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php

Feature Level Fusion of Night Vision Images Based on K-Means Clustering Algorithm

Adnan Mujahid Khan*, Bushra Kayani*, Asif M. Gillani*
* Faculty of Computer Science & Engineering, GIK Institute Topi, Swabi, Pakistan.

Abstract: **A region based visual and thermal image fusion technique based on k-means clustering algorithm is presented in this paper. This novel region fusion method segments regions of interest from thermal image using k-means clustering algorithm. Later on, these regions of interests are fused with visible image in DWFT domain. A prominent feature of our proposed technique is its near-real-time computation. Objective comparison of the scheme proposed in this paper has been done with other well known techniques. Experimental results and conclusion outlined in this paper will explain how well the proposed algorithm performs.**

Keywords: Image fusion, discrete wavelet frame transform (DWFT), k-means clustering, Discrete wavelet transform (DWT), Mutual Information (MI).

1. INTRODUCTION

For past many years, there has been an increasing interest of researchers in the area of thermal and visual image fusion, because of its applications in both civilian and military projects. The motivation behind this increasing interest is to get better situation assessment; which can never be obtained from the data acquired from a single sensor, either infra-red or visual.

Due to immense advances in sensor technology, the requirement to monitor our surrounding has been greatly increased. In various military and civilian applications, we need to fuse thermal and visible images into a single image because information from single source is not sufficient to provide a clear perception of the real world. In military applications, for example, night vision sensors are normally used for helicopter navigation and driving. Thermal sensors provide clues on the terrain and surrounding environment by sensing emitted infra-red radiation. These features can be as subtle as the cooling hand and footprints left by a person who has recently passed through the area [2].Visible image on the other hand provides information based on the reflected light. If we, in some way, combine images from both sensors, into a single image in real time, it could be used in helmet mounted display for soldiers and fire fighters, or on a penal mounted in a vehicle, which enable the driver to drive with a clear view even in bad weather conditions. In recent years, many researchers have taken keen interest in the use of thermal and visible images to detect mines [3], mapping and scene understanding [4] for military defence applications. Other civilian application areas of image fusion include medical imaging, search and rescue operation, police surveillance and fire fighting [2].

Image fusion is generally performed at three different levels of information representation; these are pixel level, feature level and decision level [5]. Fusing images at pixel level means to perform integration at a level where the pixels are least processed. Each pixel in the fused image is calculated from pixels in the source images by for-example averaging. Fusion at feature level first requires extraction of features from the source images (through e.g. segmentation); fusion then takes place based on features that match some selection criteria. At symbol level/decision level, the output from the initial object detection and classification using source images is then fed into the fusion algorithm. Every image fusion algorithm is performed at one of these three levels or some combination thereof. Algorithm proposed in this paper is based on feature level fusion, images we tend to fuse are segmented into regions and fused image is captured from the integration of required segments from both the images.

Looking in the literature, we find image fusion techniques which vary from simple pixel averaging to complex methods involving principal component analysis (PCA) [6], pyramid based image fusion [7] and wavelet transform (WT) fusion [8]. All these methods mainly fuse images on pixel level, which results in reduction of contrast and addition of artifacts. We fuse image to accelerate or improve the fusion post-processing tasks like object detection or target recognition. Region fusion helps detection of objects or regions of interest with improved confidence. Piella [9] and Zhang et al. [10] proposed region fusion algorithms in which they integrate images with the help of regions of interest.

In this paper a novel image region fusion algorithm is proposed in which images are transformed using DWFT and images are fused after regions from k-means clustering algorithm are acquired.

The subsequent sections of this paper are organized as follows. Section 2 explains the components of the proposed algorithm. Section 3 explains the method with the help of algorithm and flowchart. Section 4 explains Mutual Information (MI), an objective image fusion quality evaluation measure followed by experimental results and conclusion.

2. INGREDIENTS OF PROPOSED SCHEME

The key constituents of our scheme are discrete wavelet frame and k-means clustering. Below we have discussed some of the prospects regarding these techniques.

A. Why DWFT?

The lack of translation invariance together with rotation invariance is the key drawback of DWT in feature extraction. Due to shift variance the fusion methods using DWT lead to unstable and flickering results. This can be overcome with DWFT by calculating and retaining wavelet coefficients at every possible translation of convolution filters or in other words the redundant transforms. More detail can be found in MATLAB wavelet toolbox, where it is called Stationary Wavelet Transform (SWT).

B. k-mean clustering

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 73–76.

K-means is a technique for clustering which partitions a group of n data items into k groups and finds a cluster center in each group such that a cost function is minimized [11]. This algorithm is used for clustering because of its stability and extensibility. It's a heuristic approach of clustering under unsupervised environment.

3. Proposed Image Fusion Scheme

It is important to know for the readers that the set of images used in this algorithm are registered images. With registration we find correspondence between images. It is necessary because only after it is ensured that spatial correspondence (information from different sensors can be guaranteed to come from identical points on inspected object) is established, fusion makes sense. For image registration, normally two approaches are used. They are global [15] and local [16] motion estimation. More detail on image registration can be found in [12], [13].

A. Algorithm

1. Take DWFT of both visual and thermal image resulting into their 1 approximation and 3 detail sub-bands. The decomposition level in this case has been set to 1. This is because; this decomposition level gives optimum results.
2. Segment the thermal image into important and sub-important regions using k-means clustering algorithm, resulting into a clustered image where zero represents the unimportant region and one represents important region.
   If we visually analyze the thermal image, it contains grey levels either belonging to upper range of grey levels (e.g. greater then 200) or belonging to lower medium range of grey levels (e.g. less than 140). Exploiting this fact, if we segment these grey levels into two parts (i.e. important and sub-important regions), we can extract significant important details from thermal image which can be further used for fusion.
3. Compute the fused coefficient map using the following relation.

$$F(i,j)=\begin{cases}T(i,j)\longrightarrow if C(i,j)=1\\ V(i,j)\longrightarrow if C(i,j)=0\end{cases} \quad (1)$$

   here $F(i,j)$ (fused DWFT coefficients) is equated to $T(i,j)$ (Thermal image DWFT coefficients) if Clustered image ($C(i,j)$) is 1 at index i, j and similarly $F(i,j)$ is equated to $V(i,j)$ (visual image DWFT coefficients) when $C(i,j)$ is 0.
4. Take the inverse discrete wavelet frame transform (IDWFT) of the fused coefficient map and get the fused image.

B. Flowchart

The general framework of the proposed algorithm can be shown with the help of flowchart. Algorithmic steps performed at each major step of algorithm are shown in mathematical form in Fig.1.

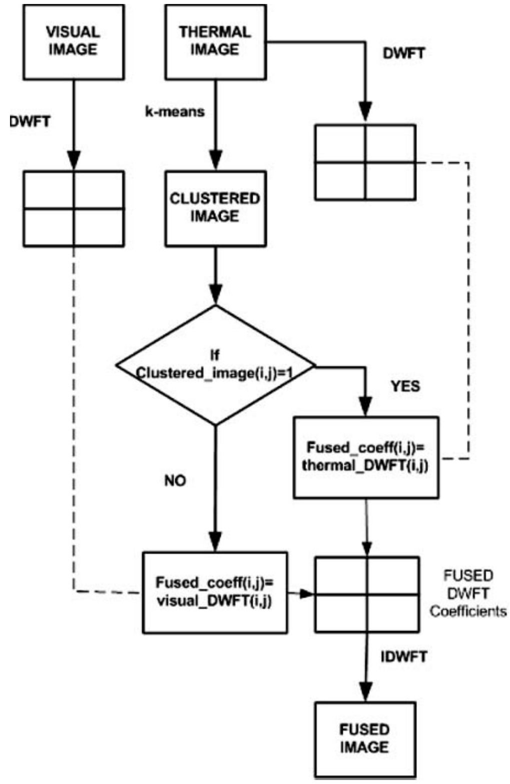


Fig. 1. Flowchart of the proposed scheme

4. Mutual Information

Mutual Information [1] has been used as a objective image fusion quality evaluation measure. Mutual Information of X and Y is the amount of information gained about X when Y is learned and vice versa. It is observed to be zero in case X and Y are independent. Mutual Information between two source images and the fused image is defined as follows.

$$I_{FA}(f,a)=\sum_{f,a}p_{FA}(f,a)\log_2\frac{p_{FA}(f,a)}{p_F(f)p_A(a)} \quad (2)$$

$$I_{FB}(f,b)=\sum_{f,b}p_{FB}(f,b)\log_2\frac{p_{FB}(f,b)}{p_F(f)p_B(b)} \quad (3)$$

In equation 2 and 3 $p_{FA}(f,a)$ and $p_{FB}(f,b)$ are the joint histograms of fused image and image A and fused image and image B. The mutual information is thus calculated as:-

$$MI_F^{AB}=I_{FA}(f,a)+I_{FB}(f,b) \quad (4)$$

The more the value of MI the better is the quality.

5. Results And Discussions

Three existing image fusion schemes are used for comparative analysis of our proposed scheme. These schemes are.

A. DWT based image fusion

In this method, images are first decomposed using DWT. Approximation and detail sub-bands are fused by choosing maximum wavelet coefficients from both the DWT coefficients of source images. Fused image is acquired by applying Inverse DWT [5].

B. aDWT based image fusion

An advanced wavelet transform (aDWT) method that incorporates principal components analysis and morphological processing into a regular DWT fusion algorithm. [6].

C. Image region fusion using fuzzy logic

In this method images are first segmented into three regions (Important, sub-important and background) using k-means clustering algorithm. These regions are then fused using fuzzy inference system [7].

In all these schemes images are decomposed to 1st level and wavelet named 'db1' is being used. This fusion algorithm is done using MATLAB.

A set of 32 UN camp visual and infra-red registered images with 256 grey levels is tested on proposed scheme and the schemes mentioned above. The results obtained from these tests are shown in the graph (Figure 3).

D. Computational Time

As mentioned in the abstract that the elapsed computational time of our proposed scheme is very less as compared to other known techniques which can also be verified by the following graph in fig. 2.

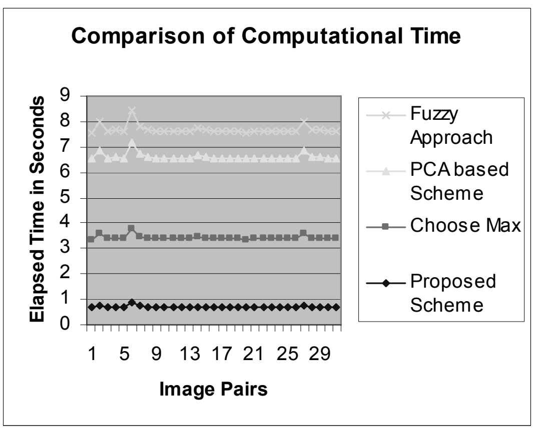


Fig. 2. Graph showing the Computational Time of proposed scheme in comparison with the computation time of other known techniques. Clearly, the proposed scheme performs quite better.

Two sample images have been taken for subjective/ visual comparative evaluation of the proposed and existing schemes. Fig. 4 and 5 are there for subjective evaluation of proposed scheme with other techniques.

6. CONCLUSION

An image fusion algorithm, based on k-means clustering and DWFT is presented in this paper. With experimental results and discussion we conclude that proposed algorithm outperforms existing fusion schemes as far as the quality of fused image is concerned and in computational time too; as this scheme fuse images in near-real time.

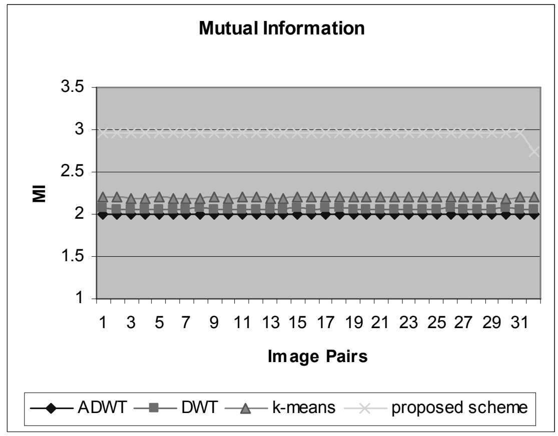


Fig. 3. Graph showing the temporal comparison of value of Mutual Information of proposed scheme with other known techniques. Again, it is evident that the proposed scheme performs far better than other given techniques.

Future work includes extending our algorithm for concealed weapon detection. In this case, thermal image provides information regarding the concealed weapon. So, all we have to do is, extract the concealed weapon information from thermal image, refine it to extract the regions of interest and then fuse it with visible image.

REFERENCES

[1] Shutao Li, James T. Kwok Yaonan Wang, Combination of images with diverse focuses using spatial frequency, *Information fusion, vol 2, 2001, pp. 169-176.*

[2] M. E. Ulag and C. L. Mc Cullough, Feature and data level image fusion of visible and infra red images, *Spie Conference on Sensor Fusion*, *Vol* 3719, Orlando, Florida, April 1999.

[3] D. W. McMichael, Data Fusion for Vehicle-Borne Mine Detection, *EUREL Conference on Detection of Abandoned Land Mines*, pp. 167-171, 1996.

[4] T.A. Wilson, S. K. Rogers, and M. Kabrisky, Perceptual-Based Image Fusion for Hyperspectral Data, *IEEE Trans. On Geoscience andRemote Sensing*, *Vol*. 35, no. 4, 1997.

[5] Gonzalo Pajares, Jesús Manuel de la Cruz, A wavelet-based image fusion tutorial. *Pattern Recognition 37*(9), 2004, 1855-1872.

[6] Yufeng Zheng, Edward A. Essock and Bruce C. Hansen, An Advanced Image Fusion Algorithm Based on Wavelet Transform – Incorporation with PCA and morphological Processing, *Proceedings of the SPIE, volume 5298*, 2004, 177-187.

[7] A.Toet, Image fusion by a ratio of low pass pyramid, *Pattern Recognition Letters, vol.9*, no.4, 1989, 245-253.

[8] H.Li, S.Manjunath and S.K.Mitra, Multi-sensor image fusion using the wavelet transform, *Graphical Models and Image Processing, vol.57*, no.3, 1995, 235-245.

[9] Piella. G., A general framework for multi-resolution image fusion: from pixels to regions, *Information Fusion,* pp.259-280, 2003.

[10] Zhang, Z., Blum, R.S., A Region-based Image Fusion Scheme for Concealed Weapon Detection. *Proceedings 31st Annual Conf. Information Sciences and Systems*, Baltimore, MD, pp.168-173. 1997.

[11] S.R. Jang, C.T. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence* (USA, Prentice Hall Inc, 1997).

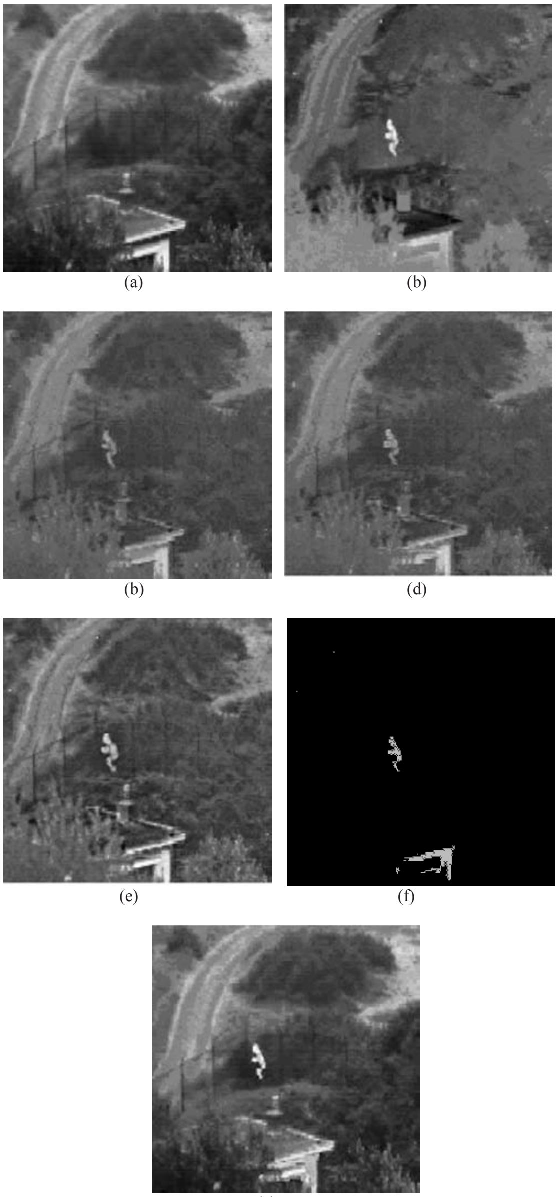


Fig. 4 Fusion of Night time images: (a) Visual Image (b) Infra-red Image
(c) DWT fusion (d) aDWT fusion (e) fuzzy logic based fusion
(f) k-mean clustered image (g) Proposed scheme

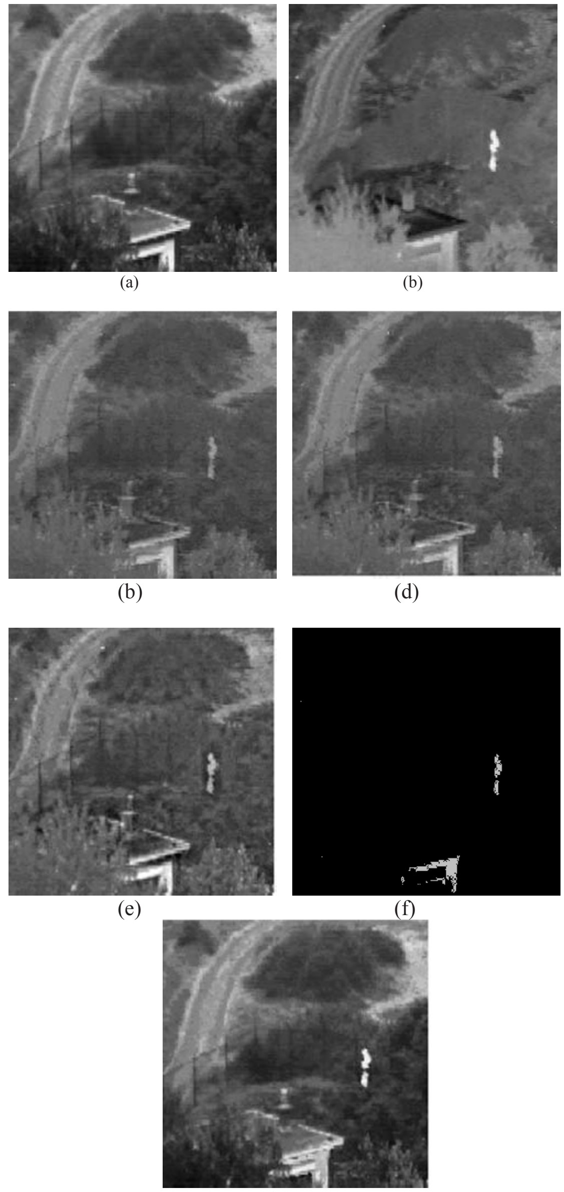


Fig. 5. Fusion of Night time images: (a) Visual Image (b) Infra-red Image
(c) DWT fusion (d) aDWT fusion (e) fuzzy logic based fusion
(f) k-mean clustered image (g) Proposed scheme

[12] R. K. Sharma and Misha Pavel, Multi-sensor Image Registration, *SID Digest Society for Information Display*. Vol xxviii, May 1997, pp.951-954.

[13] Brown, L.G, A survey of Image registration techniques*, ACM Computing Survey 24*, 1992, 325-376.

Liu Gang, Jing Zhong-liang, Sun Shao-yuan, Multi resolution image fusion scheme based on fuzzy region feature, *Journal of Zhejiang University Science A, 7*(2), 117-122.

[14] G. W. M. Hansen, K. Dana, and P. Burt, Real Time scene stabilization and motion construction, in *Proceedings of Second IEEE International Workshop on Applications of Computer Vison,* 1994, pp. 54-62.

[15] L. Bogoni and M. Hansen, Pattern Selective Color image Fusion, *Pattern Recognition,* Vol. 34, pp. 1515-1526, 2001.

Chaotic Fractals with Multivalued Logic in Cellular Automata

Amal K. Ghosh
Electronics Test and Development Centre, West Bengal
4/2 B.T. Road, Kolkata-700 056, India.
Email :amal_k_ghosh@rediffmail.com

Pabitra Pal Choudhury
Applied Statistics Unit, Indian Statistical Institute.
203, B. T. Road, Kolkata - 700 108, India
Email : pabitra_ppc@hotmail.com

Amitabha Basuray
Department of Applied Optics & Photonics University of Calcutta,
92, A.P.C.Road, Kolkata-700 009, India
Email : abasuray@rediffmail.com

Abstract: This report deals with the application of multi-valued logic in cellular automata. A four valued logic system with dibit representation has been considered in this case. The general properties and their relations to build such logical systems are also investigated and the question of implementation of this logical system in cellular automata environment has also been studied. It is shown, that chaotic fractals i.e, fractals as function of initial conditions are formed in such cases. It is interesting to note also that fractals so formed are multifractals and thus may have applications in analyzing natural fractal formation.

Subject. terms: Multivalued Logic, Cellular Automata, Fractals.

1. Introduction

Two valued logic includes those logical systems, which are based on the hypothesis of two valued propositions. This two valued conception of logic is expressed, for instance, by constructing the propositional calculus in such a way that all tautologies of the two-valued algebra of propositions are derivable in it, so that it is deductively complete with respect to the two-valued algebra of proposition. By many –valued logical systems we mean many-valued constructions in the logic of proposition and predicates. It includes along with construction of such logical systems, the investigation of their properties and relations [1]. Historically, the first many-valued propositional logic is the system constructed by Lukasiewicz[2].

Starting with the analysis of modal propositions, Lukasiewicz came to the conclusion that two-valued logic is insufficient for the description of the mutual relations and properties of these propositions that we need here a logic in which, besides the classical truth values ‘true’ and ‘false’, there is a third value ‘possible’, ‘neutral’(a neutral intermediate value). Later Lukasiewicz [3] revised his point of view on modal logic and applied instead of a three valued logic, a four valued one in which the laws of two-valued proposition logic remain valid. In the present report we have has shown its importance in application to cellular automata.

Cellular automata are mathematical idealization of physical systems in which space and time are discrete, and physical quantities take on finite set of discrete values. A cellular automation consists of a regular uniform lattice (or ‘array’), usually infinite in extent, with a discrete variable at each site (cell). The state of a cellular automation is completely specified by the variables at each site. A cellular automation evolves in discrete time steps, with the value of the variable at one site being affected by the variables at sites in its ‘neighborhood’ on the previous time step[4]. The ‘neighborhood’ of a site is typically taken to be the site itself and all immediately adjacent sites.

The variables at each site update synchronously based on the values of the variables in their neighborhood at the preceding time step and according to a definite set of ‘local rules’. The development of structure and pattern in biological systems often appears to be governed by very simple local rules and thus may be described by cellular automation model[5]. Any physical system satisfying differential equation may be approximated as a cellular automation by introducing fine differences and discrete variables. Multivalued logic may play an important role in solving such differential equations. In the present report the different possibilities of such applications of multivalued logic has been discussed.

In the past few years some work has also been initiated in the field of optoelectronic cellular automata for massive parallel processing tasks in relation to parallel computing [6-7] as well as for parallel processor for vision tasks [8]. Optics has advantages over electronics not only in parallel processing but also in four- bit representation. As in this case along with presence or absence of light the nature of the polarization state of the light beam may be an additional parameter for representing such states.

2. Multivalued Logic

2.1 Galois field of the form GF(2^m)

The field of integers modulo a prime number is, of course,

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 77–82.

the most familiar example of a finite field, but many of its properties extend to arbitrary finite fields. Let F be a field . A subset p of that is itself a field under the operations of F will be called a sub field of F and F is called an extension (field) of p. If $p \neq F$, it is said that p is a proper subfield of F. If F is a finite field such that F has p^m elements, where prime p is the characteristic of F and m is the degree of F over its prime subfield then the finite (Galois) field $GF(p^m)$, which has p^m elements, supports basic arithmetic operations under the closure condition. For quaternary logic i.e, with four states, Galoi's field may be represented as $GF(2^2)$.

However, the data representation in such cases is important. The elements of $GF(p^{2m})$ can be represented using the integers from 0 to p-1 , where p is at least two but $GF(2^m)$, the extended field can not be represented by integers alone in a straight forward way. The main conventions for representing elements of $GF(p^m)$ are the exponential format and the polynomial format. Here we present a method for constructing the Galoi's field of 2^m elements (m>1) from the binary field GF(2), (m>1) from the binary field GF(2). The basis of this representation is if all $\gamma_i \in GF(2)$ for $0 \le i \le m-1$, are linearly independent, over the set $\Gamma = \{\gamma_0, \gamma_1, \gamma_2, \gamma_3, \ldots, \gamma_{m-1}\}$ form a basis of $GF(2^m)$ can be expressed as $\alpha = \Sigma a_i \gamma_i$, where $a_i \in GF(2)$. The term a_i is represented as the i^{th} co-ordinate of a with respect to basis Γ .

Starting with two elements, 0 and 1 from GF(2) and a new symbol α representation of the Galois field for $GF(2^m)$ may be made so that we have the following set of elements on which a multiplication operation "**.**" is defined :

$$F = \{0, 1, \alpha, \alpha^2, \ldots\ldots\ldots\ \alpha^j, \ldots\ldots\}$$

the element 1 is α^0 [9].

Next we put a condition on element α so that the set F contains only 2^m elements and is closed under the multiplication "**.**" Let p(x) be a primitive polynomial of degree m over GF(2). We assume that $P(\alpha) = 0$. Since p(x) divides $x^{2^m-1}+1$, we have

$$x^{2^m-1}+1 = q(x)\,p(x) \quad \ldots\ldots\ldots\ldots\ldots (3)$$

If we replace x by α, we obtain

$$\alpha^{2^m-1}+1 = q(\alpha)\,p(\alpha) \quad \ldots\ldots\ldots\ldots\ldots\ldots(4)$$

Since, $p(\alpha) = 0$, we have

$$\alpha^{2^m-1}+1 = 0$$

Adding 1 to both sides (using modulo 2 addition), we obtain

$$\alpha^{2^m-1} = 1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5)$$

So, under this condition that $p(\alpha)=0$, the set F become finite and contains the following elements:

$$F^* = \{0, 1, \alpha, \alpha^2, \ldots\ldots\ldots, \alpha^{2^m-2}\}$$

The non-zero elements of F* are closed under the multiplication operation defined by (1). Let i and j be two integers such that, $0 \le i$, $j < 2^{m-1}$. If $i + j < 2^{m-1}$ then

$$\alpha^i \cdot \alpha^j = \alpha^{i+j} ,$$

which is obviously a non-zero element in F. If $i + j \ge 2^{m-1}$, we can express

$$i + j = (2^m-1) + r, \quad \text{where } 0 \le r < 2^{m-1} .$$

$$\text{Then } \alpha^i . \alpha^j = \alpha^{i+j} = \alpha^{(2^m-1)+r} = \alpha^{2^{m-1}} . \alpha^r = 1 . \alpha^r = \alpha^r$$

which is also a non-zero element in F*. Hence, it can be concluded that the non-zero elements are closed under the multiplication "**.**" .

Also we see that for $0 < i < 2^{m-1}$, α^{2^m-i-1} is the inverse of α^i since

$$\alpha^{2^m-i-1} \cdot \alpha^i = \alpha^{2^m-1} = 1$$

Hence, $\{1, \alpha, \alpha^2, \ldots\ldots\ldots\ \alpha^{2^m-2}\}$ represent $2^m - 1$ distinct elements.

Next we define an additive operation + on F* so that F* forms a commutative group under "+". For $0 \le i \le 2^m - 1$ we divide the polynomial x^i by p(x) and obtain the following

$$x^i = q_i(x)p(x) + Q_i(x) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

where $q_i(x)$ and $Q_i(x)$ are the quotient and the remainder respectively. The remainder $Q_i(x)$ is obviously a polynomial of degree m-1 or less over GF(2) and is of the form

$$Q_i(x) = Q_{i0} + Q_{i1}x + Q_{i2}x^2 + \ldots\ldots\ldots + Q_{i,m-1}x^{m-1} \quad \ldots\ldots\ldots(7)$$

Since x and p(x) are relatively prime i.e., they do not have any common factor except 1, x^i is not divisible by p(x). Thus elements in the $GF(2^m)$ are $\{0, 1, \alpha, \alpha^2, \ldots\ldots\ldots \alpha^{2^m-2}\}$ where α is the root of m degree primitive polynomial.

If m=2 then $2^m - 2 = 2$.

Hence $\{0,1, \alpha, \alpha^2\} \equiv \{0,1, \alpha, \alpha+1\}$ are the elements in quadruple logic of 2^2 states.

Thus elements of $GF(2^2)$ are

	Ordered pairs
$0 = 0 + 0.\alpha$	0 0
$1 = 0.\alpha + 1$	0 1
$\alpha = \alpha + 1.0$	1 0
$\alpha^2 = \alpha + 1$	1 1

Ordered pairs represent the elements and the states {0,1,2,3} are represented by dibit as {00, 01, 10, 11} respectively.

2.2 Truth Tables based on dibit representation

The basic logical operations with dibit representation as mentioned in the Sec.2.1 may be expressed in the following fashion. In the present system the normal logical gates e.g., OR, AND, NOT or X-OR may be represented bit wise. The truth table for these conventional bit wise logic gates are represented in Table – I.

B \ A	00	01	10	11
00	00	01	10	11
01	01	01	11	11
10	10	11	10	11
11	11	11	11	11

(a)

B \ A	00	01	10	11
00	00	00	00	00
01	00	01	00	01
10	00	00	10	10
11	00	01	10	11

(b)

A	A′
00	11
01	10
10	01
11	00

(c)

B \ A	00	01	10	11
00	00	01	10	11
01	01	00	11	10
10	10	11	00	01
11	11	10	01	00

(d)

Table-I: Truth tables for (a) OR, (b) AND, (c) NOT and (d) XOR using bitwise Logic.

At this point it is interesting to note that the addition and multiplication are not simple bit-wise XOR and AND operations, these operations are performed in bit serial fashion. This is apparent from the truth table given in Table-I. In binary system the XOR gate is also the modulo-2 gate and thus gives the addition which is not true in case of dibit logic gates based on binary logic for each bit as in such cases, XOR operation is not the modulo-4 gate. Similarly, AND gate defined in Table-I does not represent either the multiplication logic or generates the carry bit. Then four-valued logic system calls for a more number of gates and the mathematical equations are to be developed using bit serial fashion. The most important mathematical gates i.e., the addition gate and the multiplication gates may be defined in the following fashion.

For addition gate, if

$$a_j\, a_i + b_j\, b_i = c_j\, c_i$$

then $c_i = a_i \text{ XOR } b_i$

and $c_j = (a_i \text{ AND } b_i) \text{ XOR } (a_j \text{ XOR } b_j)$

where " + " stands for addition.

Similarly, the multiplication gate

$$a_j\, a_i \,.\, b_j\, b_i = d_j\, d_i$$

then $d_i = a_i \text{ AND } b_i$

and $d_j = (a_i \text{ AND } b_j) \text{ XOR } (a_j \text{ AND } b_i)$

The corresponding truth tables for the addition and multiplication are given in Tables II(a) and II(b) respectively.

$b_j\, b_i$ \ $a_j\, a_i$	00	01	10	11
00	00	01	10	11
01	01	10	11	00
10	10	11	00	01
11	11	00	01	10

(a)

$b_j\, b_i$ \ $a_j\, a_i$	00	01	10	11
00	00	00	00	00
01	00	01	10	11
10	00	10	00	10
11	00	11	10	01

(b)

Table-II: Truth tables for (a) Addition and (b) Multiplication.

Thus the logical and mathematical operations over $GF(2^m)$ field can be subdivided in different classes. Some operations are bit-wise but for others it is in bit serial fashion or in which some information are carried over from the results of earlier bit.

3. Application of digital logic to cellular automata

As mentioned earlier, cellular automata are mathematical idealization of physical systems in which space and time are discrete, and physical quantities take on a finite set of discrete values. Wolfram[4] presented a detailed analysis of elementary cellular automata consisting of a sequence of sites with value 0 or 1 on a line, with each site evolving deterministically in discrete time steps according to definite rules involving the values of the nearest neighbors. He has also shown that an eight-bit number describes the local rules for a one dimensional neighborhood-three cellular automation. Hence, there are $2^8 = 256$ possible distinct automation rules in one dimension. Wolfram also showed that of these only 32 rules may be considered to be legal.

If now one finds the evolution of all 32 possible legal cellular automata from an initial configuration containing a single site with value 1, it is interesting to note that only nine rules show some kind of fractal formulation i.e, a particular configuration repeats itself. The polynomial rules for said rules are

(1) $S_{n-1}S_n \oplus S_nS_{n+1} \oplus S_{n-1}\oplus S_{n+1}$ rule no.18
(2) $S_{n-1}S_n\, S_{n+1} \oplus S_{n-1} \oplus S_n \oplus S_{n+1}$ rule no.22
(3) $S_{n-1}S_{n+1} \oplus S_{n-1} \oplus S_n \oplus S_{n+1}$ rule no.54
(4) $S_{n-1} \oplus S_{n+1}$ rule no.90
(5) $S_{n-1}S_n \oplus S_nS_{n+1} \oplus S_{n-1}\, S_{n+1} \oplus S_{n-1} \oplus S_n \oplus S_{n+1}$ rule no.126
(6) $S_{n-1}S_n\, S_{n+1} \oplus S_{n-1}S_n \oplus S_n\, S_{n+1} \oplus S_{n-1} \oplus S_{n+1}$ rule no.146
(7) $S_{n-1} \oplus S_n \oplus S_{n+1}$ rule no.150
(8) $S_{n-1}S_n\, S_{n+1} \oplus S_{n-1}S_{n+1} \oplus S_{n-1} \oplus S_n \oplus S_{n+1}$ rule no.182
(9) $S_nS_{n+1} \oplus S_{n-1} \oplus S_{n+1}$ rule no.210

S_{n-1}, S_n and S_{n+1} are the values of left number, the number under consideration and its right number and $\oplus$ stands for Ex-OR operation. No fractal formation is taking place using other possible rules. These cellular automation rules exhibit the important simplifying feature of "additive superposition" or additivity. Evolution according to such rules satisfies the superposition principle.

$$S_o = a_o \oplus t_o \leftrightarrow S_n = t_n \oplus u_n$$

The additive principle of nine rules mentioned above combine values at different sites by addition modulo 2. Thus it is evident that modulo-2 operation or Ex-OR logic play an important role in cellular automata.

4. Application of four-valued logic to cellular automata rules

The multivalued GF(2^2) cellular automata can be viewed as an extension of GF(2) CA discussed in earlier section. It consists of an array of cells, spatially interconnected in a regular manner, each cell is capable of storing an element of GF(2^m). In effect each GF(2^2) CA cell has the capability to store 0 ,1, 2 or 3.

In this paper we will discuss the basic three neighbor rules same as that used in binary system. However, the rules may be divided in two classes, one in which bit wise operation is made and in other groups when the logic operation is made in bit serial fashion.

In case of bit wise logic it is evident that only in nine rules as in case of binary system show fractal formation. However, it is to be noted that each neighbouring state i.e, S might have different values depending upon the Gray levels, in the present case which is four. It is also verified that in such cases exclusively the nine rules, which lead to fractal formation in binary system, are giving rise to fractal formation. The rule representing are equivalent form of the rule 90 in binary logic defined as

$$S_n(t+1) = S_{n-1}(t) \oplus S_{n+1}(t)$$

where $\oplus$ stands for XOR operation.

The evolution process based on XOR with different values at the initial states are shown in Figs..1(a to c). It is interesting to note that the nature of the fractals in all cases are same only the value at each site is higher or lower according to the initial state. This is expected and will be evident as X-OR is inferred as a linear gate. As this operation is linear, hence evolution process based on Ex-OR logic should not depend on the initial state. At this point we recall that addition logic rules give rise to fractal formation in binary logic, however, XOR operations are not the addition logic in multivalued system. So we use an equivalent addition rule in multivalued system as

$$S_n(t+1) = S_{n-1}(t) + S_{n+1}(t)$$

- where '+' stands for additive logic.

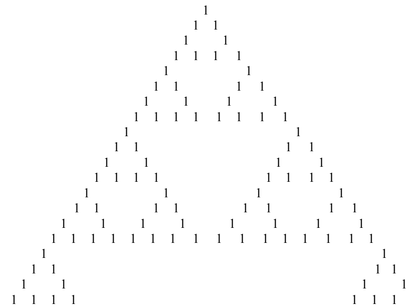


Fig.1(a): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 90 for the initial state 01 → 1

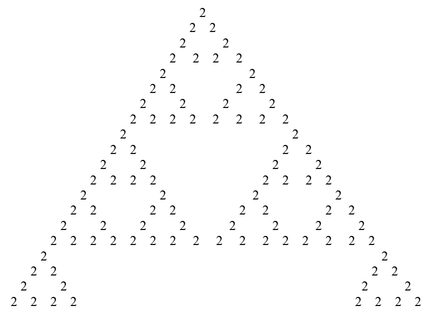


Fig. 1(b): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 90 for the initial state 10 → 2

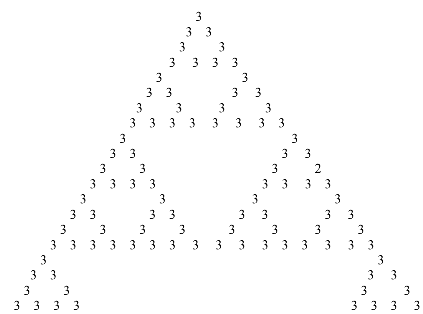


Fig. 1(c): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 90 for the initial state 11 → 3

Here also, it is interesting to note that the nine rules mentioned earlier replacing XOR logic with additive logic, give rise to fractal formation. The evolution process based on this logic rule with different initial states are given in Figs. 2 (a to c).

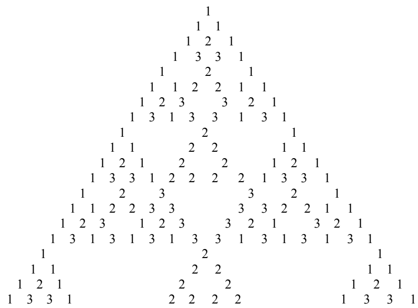


Fig.2 (a): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 90 for the initial state 01 $\rightarrow$ 1

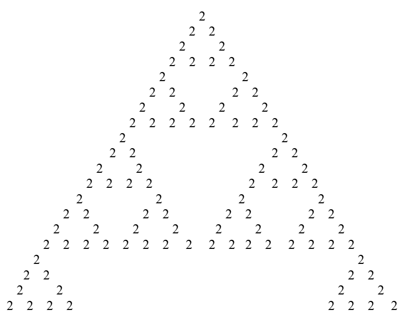


Fig.2 (b): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 90 for the initial state 10 $\rightarrow$ 2 .

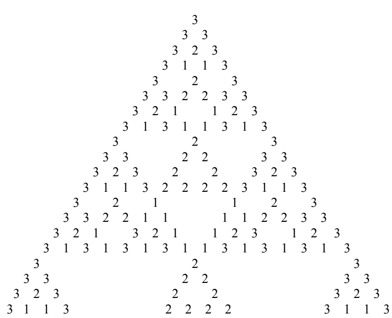
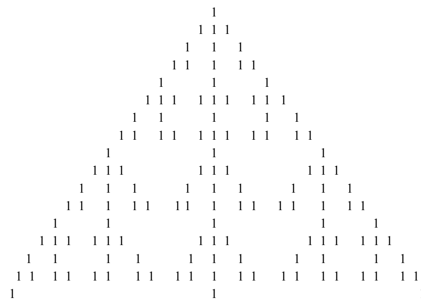


Fig.2 (c): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 90 for the initial state 11 $\rightarrow$ 3.

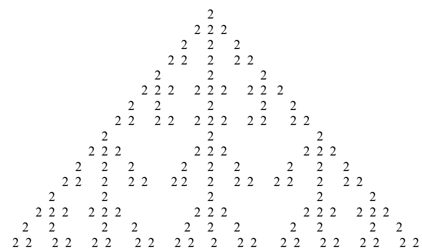
It is interesting to note that the nature of the fractal forms becomes a function of initial condition. The multivalued logic opens the horizon of non-linear equation simulation and fractal formation. The similar results with a rule equivalent to Rule 150 of binary system are also shown in figure 3(a to c) for XOR and figure 4(a to c) for additive gates with different initial states using the following rules

$$S_n(t+1)= S_{n-1}(t) \oplus S_n(t) \oplus S_{n+1}(t)$$

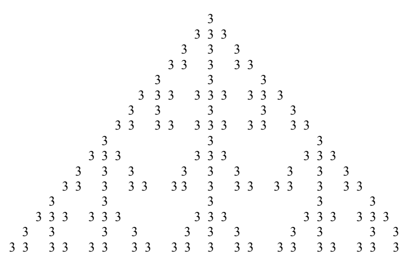
and $$S_n(t+1)= S_{n-1}(t) + S_n(t) + S_{n+1}(t)$$



3 (a): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 150 for the initial state 01 $\rightarrow$ 1



3 (b): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 150 for the initial state 10 $\rightarrow$2



3 (c): Evolution of Cellular automata using bitwise Ex-OR equivalent to Rule 150 for the initial state 11 $\rightarrow$ 3.

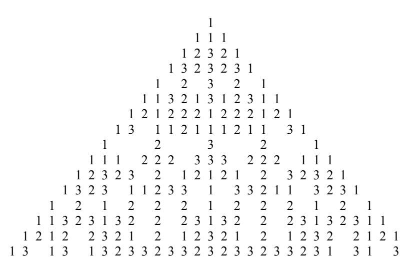


Fig. 4 (a): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 150 for the initial state 01 $\rightarrow$ 1.

```
                              2
                            2 2 2
                          2   2   2
                        2 2   2   2 2
                      2       2       2
                    2 2 2   2 2 2   2 2 2
                  2   2       2       2   2
                2 2   2 2   2 2 2   2 2   2 2
              2               2               2
            2 2 2           2 2 2           2 2 2
          2   2   2       2   2   2       2   2   2
        2 2   2   2 2   2 2   2   2 2   2 2   2   2 2
      2       2               2               2       2
    2 2 2   2 2 2           2 2 2           2 2 2   2 2 2
  2   2       2   2       2   2   2       2   2       2   2
2 2   2 2   2 2   2 2   2 2   2   2 2   2 2   2 2   2 2   2 2
```

Fig. 4 (b): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 150 for the initial state $10 \rightarrow 2$

```
                              3
                            3 3 3
                          3 2 1 2 3
                        3 1 2 1 2 1 3
                      3   2   1   2   3
                    3 3 1 2 3 1 3 2 1 3 3
                  3 2 3 2 2 2 3 2 2 2 3 2 3
                3 1   3 3 2 3 3 3 2 3 3   1 3
              3       2       1       2       3
            3 3 3   2 2 2   1 1 1   2 2 2   3 3 3
          3 2 1 2 1   2   3 2 3 2 3   2   1 2 1 2 3
        3 1 2 1   3 3 2 1 1   3   1 1 2 3 3   1 2 1 3
      3   2   3   2   2   2   3   2   2   2   3   2   3
    3 3 1 2 1 3 1 2   2   2 1 3 1 2   2   2 1 3 1 2 1 3 3
  3 2 3 2   2 1 2 3   2   3 2 1 2 3   2   3 2 1 2   2 3 2 3
3 1   3 1   3 1 2 1 1 2 1 1 2 1 2 1 1 2 1 1 2 1 3   1 3   1 3
```

Fig.4 (c): Evolution of Cellular automata using addition modulo 4 logic equivalent to Rule 150 for the initial state $11 \rightarrow 3$

It shows that in binary field the additive logic is responsible for the fractal formation, whereas in multivalued extended field the additive logic gives chaotic fractals. A close look to these fractals will reveal that they are multifractals, one important aspect of such studies is that the natural fractals are also multifractals [10], hence fractal geometry using multivalued logic may play an important role in natural fractal studies.

4. Conclusion

Over the last two decade, physicist, biologist, astronomers and economists have created a new way of understanding the growth complexity in nature. One of the reason behind such idea is that the structural determinism and apparently accidental development are not mutually exclusive, but rather their co-existence is more the rule in nature. It has been seen that the development of a process over a period of time and the structural process of such developments may be explained in many cases using chaos theory and fractal geometry. At this point, additive digital logic in cellular automata environment has potential application possibilities in different scientific fields. A good review of such applications is given in reference [13].

The use of multivalued logic in digital logic architecture will extend such application possibilities to more involved problems through chaotic fractals. The application of variable dimensional Galois fields in computational architecture was discussed by Hasan and Ebledaei [14]. Hasan and Warsal [15] have also suggested design of a possible instruction set architecture for different cryptographic applications. This may be considered as a new emergent field with application to not only in analyzing the natural structures encountered in biology but also in future architecture of computer, communication and image processing.

In optics, two orthogonal states of polarization as well as that of absence and presence of light may express the binary states. Using both the properties at a time we can generate four-state logic system using dibit representation. In the two previous papers [11,12] Basuray and others have used such representation to trinary logic systems. In this report the logic used has been extended to incorporate quadruple logic systems.

References

1. Zinovev, A, A : Philosophical problems of many valued logic [A revised edition] edited and translated by Guido Kung and David Dinsmore comey, D. Reidle publishing company dordrecht-Holland,(1963).
2. Lukasiewicz , J : O logice trojwartosciowej, Ruch filozoficzny,5,169-170 (1920).
3. Lukasiewicz, J: Aristotles Syllogistic from the stand point of modern formal logic, Oxford, London, 1951.
4. Wolfram Stephen : Statistical mechanics of cellur automata, Review of mod. Phys. 55 pp. 601-694 (1983).
5. Stanffer Dietrich : Cellur Automata from the book Fractals and Disordered Systems edited by Arumin Bunde and Sholmo Havlin, Springer-Verlag, Berlin Heidelberg 298-321(1991).
6. J.Taboury, J.M.Wang, P.Chavel, F.Devos and P.Garda, Optical Cellular processor architecture 1: Principles , *App. Opt*. Vol 9, pp 1643-1650(1988).
7. J. Taboury, J. M. Wang, Pchavel, F. Devos Optical Cellular processor architecture, 2: Illustration and system considerations , *App. Optics*, Vol 28, pp 3138(1989).
8. P. Chapel, A Cavrineli and I. Glasv, Opto electronic cellular automata for video real time vision *Optics in computing OC*" 2000 Canada, Ouebec 18-23 June 2000, SPIE, 2000 pp 374-381(2000).
9. Lin, Shu and Costello, D. J. JR. "*Error Control Coding: Fundamentals and Applications* ", Prentice –Hall, New Jersy (1983).
10. Stanley, H.E : Fractals and multifractals: The interplay of Physics and Geometry from the book '*Fractals and Disordered Systems*' edited by Arumin Bunde and Sholmo Havlin, Springer-Verlag, Berlin Heidelberg 1-50(1991).
11. Amal K. Ghosh and A. Basuray, "Trinary optical logic processors using shadow casting with polarized light", *Optics Communications*, vol.79,number 1,2 , October 1990, pp 11 – 14.
12. A. Basuray, S. Mukhopadhyay, Hirak K. Ghosh, A. K. Dutta, " A tristate optical logic system", *Optics Comm*.,vol.85, Sept.1991, pp 167 -170.
13. P. Pal Chaudhury, D. R. Chowdhury, S. Nandi and S. Chatterjee " *Additive Cellular Automata – Theory and applications:*Vol.1",IEEE Computer press, California, USA(1997).
14. M. A. Hasan and M. Ebledaei "Efficient architectures for computations over variable dimensional Galois fields", IEEE Trans. Circuits and Systems – I Fundamental theory and applications, Vol.45 no. 11 1205-1211(1998).
15. M. A. Hasan and A. G. Warsal "VLSI algorithms, architectures and implementation of a versatile GF(2^m) processor ", IEEE Trans. on Computers, Vol.49, 1064-1073(2000).

Semantics for an Asynchronous Message Passing System

A.V.S. Rajan, S. Bavan, G. Abeysinghe
School of Computing Science
Middlesex University
The Borroughs, London NW4 4BT UK

Abstract-**The main focus of this paper is to define the operational semantics for the message passing strategy called Asynchronous Message Passing System (AMPS) used in the distributed programming language, LIPS (Language for Implementing Parallel/distributed Systems). AMPS is a point-to-point message passing system that does not use any message buffers. It is based on simple architecture and interfaces. In order to adequately provide implementation information for the message passing strategy, we have defined the operational semantics and the codes needed for the abstract machine of LIPS.**

Keywords: Operational semantics, Asynchronous Message Passing, point-to-point message passing, abstract machine.

I. Introduction

Operational semantics describes the executional behaviour of a programming language and gives a computational model for the programmers to refer to. This paper presents the operational semantics that models the asynchronous message passing behaviour of LIPS, a Language for Implementing Parallel/distributed Systems [1].

LIPS has the following properties:

- Communication by assignment
- Separation of communication from computation
- A data flow nature coupled with asynchronous message passing.
- Portability
- A program is made up of processing nodes (processes) which are linked by unidirectional data channels that carry messages between cooperating nodes.

The detailed explanation of LIPS can be found in [1].

A LIPS program consists of a network of nodes described by a network definition and node definitions. A network definition describes the topology of the program by naming each node (representing a process) and its relationships (in terms of input and output data) to other nodes in the system. A node consists of one or more guarded processes which perform computations using the data that arrive as input and produces output that are sent to other relevant nodes. We formally specify the message passing between the various nodes in LIPS using the Specification of Asynchronous Communication System (SACS) [2], a synchronous variant of Synchronous Calculus of Communicating System (SCCS) [3] (network definition in a LIPS program). The design techniques of SACS allow the programmer to develop programs that are virtually free of livelock and deadlock conditions.

The main focus of this paper is to define the operational semantics for the message passing strategy called Asynchronous Message Passing System (AMPS) [4] used in LIPS. AMPS is a point-to-point message passing system that does not use any message buffers and is based on a simple architecture and interfaces.

The operational semantics of LIPS is described using an evaluation relation $(P, s) \Downarrow (P', s')$, where a program expression, P in state s is evaluated to P' with a change of state to s'. The particular style of operational semantics and the abstract machine we have adopted was inspired by Crole [5]. This style uses an evaluation relation to describe the operational semantics by showing how an expression evaluates to a result to yield a change of state and a compiled Code Stack State (CSS) machine for an imperative toy language called IMP.

We have defined the abstract machine for LIPS called LIPS Abstract Machine (LAM) that executes instructions using re-write rules. A re-write rule breaks the execution step into number of sub-steps and transform the given expression P into a final value V ($P \Downarrow^e V$) as follows:

$$P_0 \mapsto P_1 \mapsto P_2 \mapsto \ldots \mapsto V$$

In order to define the AMPS using the LAM, we need a few preliminary definitions. The LAM consists of rules for transforming the LAM configurations. Each configuration in the LAM is a triplet, (C,S,s) where

- C is the Code to be executed
- S is a Stack which can contain a list of integers, real numbers, Booleans, characters, or strings
- s is a state which is the same as that defined in the LIPS operational semantics.

The remainder of this paper is structured as follows: Section II describes AMPS, the architecture of the virtual message passing system which has been developed to pass messages asynchronously without message buffers. Section III describes the operational semantics and the codes needed for the AMPS. Section IV concludes the discussion.

II. The AMPS of LIPS

The Asynchronous Message Passing System (AMPS) of LIPS has been developed in order to achieve asynchronous message passing effectively across different platforms without any message buffers. AMPS consists of a very simple Data

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 83–88.

Structure (DS) and a Driver Matrix (DM). The LIPS compiler automatically generates the DS, DM and required AMPS interface codes. With network topology and the guarded processes, it is easy to identify the variables participating in the message passing. This section first describes the DS and the DM and then goes on to describe how AMPS transfers data to and from the nodes using simple interfaces.

A. *The Data Structure (DS) of AMPS*

The Data Structure is a doubly linked list where all the nodes in the network including the host node are linked to the other nodes. Each node has the following six components:

1. A node number (**NodeNum** – *an integer*) – a unique number is assigned to each node.
2. Name of the function it is executing (**name** – *a symbolic name*)
3. A pointer to the next node in the system
4. Further two pointers:
   i. A pointer to a list of input channel variables associated with that node.
   ii. A pointer to a list of output channel variables associated with that node.
5. Input channel variable – each input channel variable consists of a data field giving the channel number (**vnum** – *an integer*) and two pointers, one pointing to the next channel variable in the list and the other points to a record with the following fields
   i. Channel name (**var1**-*symbolic name*). This is used for debugging purposes.
   ii. Currency of the data present – old data (status = 0) or new data (status = 1). Only data with status = 1 is passed on to a node for processing.
   iii. Value of the data. (**value** – *actual value of specified type*)
6. Output channel variable - each output channel variable consists of a data field giving the channel number (**vnum** – *an integer*) and two pointers, one pointing to the next channel variable in the list and the other points to a record with the following fields:
   i. Channel name (**var1**-*symbolic name*)
   ii. The number of nodes that are to receive the data (**counter** – *0..n*), which will be decremented as a copy of the data is transferred to a destination node. New data is only accepted (written) when this counter is 0.
   iii. Value of the data. (**value** – *actual value of specified type*).

B. *The Driver Matrix (DM) of AMPS*

The DM facilitates the distribution of messages and contains the details of the channel variables in the network which are as follows:

i. The channel number, vnum (Integer),
ii. The node number (Integer) from where the channel variable originates (Source node),
iii. The data type of each channel variable (Integer value – 0 to 8) and
iv. The nodes where they are sent as input, the destination nodes (either 1 or 0 in the appropriate column). A '1' in a column indicates that the corresponding destination receives a copy of the input and a '0' otherwise.

All the values in the matrix are integers. The integer values given to the source nodes and destination nodes are same as the node numbers used in the DS.

C. *The Operation of AMPS*

When a node outputs a message, a message packet in the following format is generated.

`Src_Node_Number`	`Vnum`	`Type`	`data`

Message packet – 1 sent from a node

Once a piece of data is ready, the process in the source node makes the following call to AMPS:

```
Is_ok_to_send(Src_node_number, vnum)
```

When this call is received, the AMPS checks the DS to see if the output channel of the node has its copy value set to zero. If it is set to zero, it returns a value 1 else a 0. If the value received is 0, the sending process waits in a loop until a 1 is received. When a value 1 is received, the sender node sends the message in a packet using the following call.

```
Send(src_node_number, vnum, type, data)
```

On the receipt of this packet, the AMPS checks the DS to see whether the **`vnum`** and the **`type`** are correct, stores the data in the appropriate field. The copy counter is set to the number of nodes that are to receive the data by consulting the DM. The **`Send`** function returns a 1 to indicate success else a 0 to indicate a failure.

After storing the data, the AMPS consults the DM, distributes the data to other DS nodes and decrements the copy counter accordingly. Here the data is written to the input channel variable of a receiving DS node, provided the status counter of that input channel variable is 0 (that is, the channel is free to receive new data). Once the data is received the status is set to 1. If any of the DS destination nodes were unable to receive the new data, AMPS periodically checks whether they are free to accept the data.

When a guard in a node requires an input, it makes the following call to the AMPS:

```
Is_input_available(Rec_node_number, vnum)
```

The AMPS checks the appropriate DS node and the channel variable number, **`vnum`**. If the status is 1, the function returns a `'1'` to tell the caller that data is available, else it returns a 0. In the event of a `'0'` return, the DM is consulted to find the source. If the data is available, the system transfers a copy to the input channel of the waiting DS node and waits for the corresponding process to make another call of **`Is_input_available`** function.

If the receiving call gets a `'1'` in return, then the node makes a request to AMPS to send the data. The AMPS extracts the data from the appropriate channel of the DS and returns it to the calling process and sets the status to 0. The data is sent in the following format:

Node Number	Vnum	Type	data

Message Packet-2 sent from AMPS

If a 0 is returned to the **`Is_input_available`** function, then the process does other things or repeat similar calls.

III. Operational Semantics For The AMPS

In this section, we describe the operational semantics for the AMPS by defining the following:

- Primitives needed for the message passing
- Basic rules for the communication between various process nodes and
- Evaluation relation for these rules.

We assume that the assignment statement, while-statement, and if-statement have already been defined.

A. The Primitives of Asynchronous Message Passing and Their Semantics

LIPS has been extended with certain data types and functions, in addition to the existing data types and commands, to handle the communication between processes. TABLE I lists the additional data types of LIPS.

TABLE I
DATA TYPES FOR AMPS

Name of the type	*Purpose*
`channel`	a means of communication that carries data belonging to allowed types.
`Flag`	which can be set or reset depending upon the availability of data type = binary.
`node_number`	number of the node from where the data is received or to where the data is sent
`Vnum`	variable number – a unique number is assigned to the channel variables in the network
`type`	is a unique number assigned to each of the data types in LIPS integer, 1 .. 9
`Data`	the data participating in the message passing which is in the string form
`data_packet`	used to pass the message between the nodes and between the data structure and data matrix. The data packet is an enumerated data type which is a combination of (node_number, vnum, type_number, data)

TABLE II lists the commands to perform asynchronous message passing.

TABLE II
COMMANDS FOR ASYNCHRONOUS MESSAGE PASSING

Name of the type	Purpose
`Is_ok_to_send (Src_node_number, vnum)`	Sender checks whether it can send data to the data structure
`Is_input_available (node_number, vnum)`	Receiver checks the availability of data
`Send(data_packet)`	Sender send the data packet

B. Communications Axioms for the Guarded Processes

As described in Section I, a node in a LIPS program is a collection of guarded processes and a guarded process consists of a guard and a statement block. A guard is a collection of channels which holds valid data for the statement block to be executed. As message passing takes place through these channels, it is appropriate to define the communication rules required for them in a guarded process.

A typical node in LIPS has the following structure:

```
node area (int x[n], int y[m]) -> (int xx, int yy){
   int r, s;
   [x[0..n1], y[0]] =>{    :
        <statements>      //process body
         :
         xx = val1;
    }
   [x[n1+1..n], y[1..m]] =>{  :
        <statements>
        :
        yy = val2;
   }
}
```

where area is the name of the node and x[i], y[i] and xx, yy are input and output channels respectively. When input channels x[0..n1], and y[0] have new data in them, the corresponding process is executed and val1 is put on the output channel xx. Now, we go on to define the communication rules required for the execution of the guarded processes.

Let GP be the set of n number of guarded processes in a process node where,

$$GP = \{gp_1, gp_2, gp_3, \dots, gp_n\}$$

Let $G_1, G_2, G_3, \dots, G_n$ be the guards corresponding to the guarded process $gp_1, gp_2, gp_3, \dots, gp_n$ respectively.

Let $G_i = \{ch_{i1}, ch_{i2}, ch_{i3}, \dots, ch_{im}\}$
where $ch_{i1}, ch_{i2}, ch_{i3}, \dots, ch_{im}$ be some channels which are waiting for the data $\forall i : 1 \leq i \leq n$ and m varies between 0 and p.

Let $fch_{i1}, fch_{i2}, fch_{i3}, \dots, fch_{im}$ be the corresponding flags associated with them $\forall i : 1 \leq i \leq n$ and m varies between 0 and p.

Flag fch_{ij} will be set to true if data is available in ch_{ij} for some i where $\forall$ i:$1 \leq i \leq n$ and j where $\forall$ i: $1 \leq i \leq m$. The code to execute the set of guarded processes in a node can be given as:

```
while(true) do{
  number = random()* no_of_guards
  if(number == guardno1) ∧ (G1){statement_block_1}
  else
  if(number == guardno2) ∧ (G2){statement_block_2}
  else
   ⋮
  else
  if(number == guard_non) ∧ (Gn){statement_block_n}
}od
```

where

- `Number` - a random integer which ranges between 1 and number of guards (no_of_guards).
- guard_no$_i$ - the number assigned to a guard (i is an integer where $1 \le i \le n$).
- the `statement_block_1` for G_1 will be executed only when data is available in the channels. If fch_{11}, fch_{12}, fch_{13}, …, fch_{1m} are the flags to the input channels for G_1, then $fch_{11} \wedge fch_{12} \wedge fch_{13} \wedge \ldots \wedge fch_{1m}$ should be true to confirm the availability of data in the input channels.
- the `statement_block_2` for G_2 will be executed only when data is available in all the channels. If fch_{21}, fch_{22}, fch_{23}, …, fch_{2m} are the flags to the input channels for G_2, then $fch_{21} \wedge fch_{22} \wedge fch_{23} \wedge \ldots \wedge fch_{2m}$ should be true to confirm the availability of data in the input channels and so on.

Let $P_{i1}; P_{i2}; P_{i3}; \ldots; P_{ik}$ be the sequence of program expressions associated with the i^{th} guarded processes where $0 \le i \le n$ and $0 \le k \le s$ for some integer s.

The communication rules needed for the AMPS are stated below:

(1) The statement for a guarded process gp_i is an alternate construct in the node expression consisting of n guarded processes. This can be stated as:

$$\boxed{\begin{array}{l} \texttt{if}(fch_{i1} \wedge fch_{i2} \wedge fch_{i3} \wedge \ldots \wedge fch_{im}) \\ \qquad\qquad \texttt{then } P_{i1}; P_{i2}; P_{i3}; \ldots; P_{ik} \end{array}}$$

where $P_{i1}; P_{i2}; P_{i3}; \ldots; P_{ik}$ be the sequence of program expressions and they will be executed for gp_i only when $fch_{i1} \wedge fch_{i2} \wedge fch_{i3} \wedge \ldots \wedge fch_{im}$ becomes `true` to confirm the availability of data in the input channels $ch_{i1}, ch_{i2}, ch_{i3}, \ldots, ch_{im}$ for $0 \le i \le n$ and $0 \le m \le p$ for some integer p. Whenever a guard needs data, it will make a call to the data structure to find whether data is available.

(2) The function to check whether input data is available can be stated as:

$$\boxed{\texttt{Is\_input\_available(node\_number, vnum)}}$$

(3) The function to find whether data can be sent or not can be stated as:

$$\boxed{\texttt{Is\_ok\_to\_send(Src\_node\_number, vnum)}}$$

(4) The function for the sender to send the data can be stated as:

$$\boxed{\texttt{Send(data\_packet)}}$$

The same `Send` function is to send data to a requested node.

These four rules, based on the working of AMPS form the major communication rules used in the LIPS language. The major advantage of this approach is that the user need not be concerned about the implementation and working of AMPS.

C. *Evaluation Relation for the Communication Axioms*

The types in LIPS stated are as follows:

$$\begin{array}{l} \sigma ::= \texttt{int} \mid \texttt{real} \mid \texttt{bool} \mid \texttt{string} \mid \texttt{char} \mid \texttt{channel} \\ \qquad \mid \texttt{flag} \mid \texttt{node\_number} \\ \mid \texttt{vnum} \mid \texttt{type\_number} \mid \texttt{data\_packet} \mid \texttt{cmd} \end{array}$$

TABLE III lists only the syntactic categories of AMPS of LIPS.

TABLE III
SYNTACTIC CATEGORIES FOR AMPS

Description	Category
Set of channels – positive integer values.	$\texttt{CHANNEL} \underline{\underline{\texttt{def}}} \{ch_1, ch_2, \ldots, ch_n\}$
Set of flags which takes Boolean values.	$\texttt{FLAG} \underline{\underline{\texttt{def}}} \{fch_1, fch_2, \ldots, fch_n\}$
Set of node numbers (node number is a unique number assigned to a node in a network).	$\texttt{NODE\_NUMBER} \underline{\underline{\texttt{def}}} \{\texttt{finite set of integers}\}$
Finite set of integers (the unique number assigned to a the channel variables in a network).	$\texttt{VNUM} \underline{\underline{\texttt{def}}} \{\texttt{finite set of integers}\}$
Finite set of integers – number assigned to each type of data	$\texttt{TYPE} \underline{\underline{\texttt{def}}} \{1,2,3,4,5,6,7,8,9\}$
The data in string form	$\texttt{DATA} \underline{\underline{\texttt{def}}} \{\texttt{data} \mid \texttt{data} \in \textbf{STR}\}$
Data packet	$\texttt{DATA\_PACKET} \underline{\underline{\texttt{def}}} \{\texttt{node\_number}, \texttt{vnum}, \texttt{type\_number}, \texttt{data}\}$
Function call to check for the availability of data	`Is_input_available(node_number, vnum)`
Function call to check whether can send data	`Is_ok_to_send(Src_node_number,vnum)`
Function call to send data	`Send(data_packet)`

The evaluation relation needed for the communication rules described are given below:

(1) A channel `ch` of type `CHANNEL` will evaluate to a channel in the same way an integer number gets evaluated to an integer. The deduction tree for ch is given as below:

$$\frac{}{(\underline{ch}, s) \Downarrow (\underline{ch}, s)} \Downarrow \texttt{CHANNEL}$$

The evaluation rules for flags (fch), node number (n), variable number (n), type (n) and data (data) are similar to the evaluation relation of a channel given above. They are expressed as below:

Flag (fch) $$\frac{}{(\underline{fch}, s) \Downarrow (\underline{fch}, s)} \Downarrow \texttt{FLAG}$$

Node number (n) $$\frac{}{(\underline{n}, s) \Downarrow (n, s)} \Downarrow \texttt{NODE\_NUMBER}$$

Variable number (n) $\dfrac{}{(\underline{n},\ s) \Downarrow (\underline{n},\ s)} \Downarrow \texttt{VNUM}$

Type number (n) $\dfrac{}{(\underline{n},\ s) \Downarrow (\underline{n},\ s)} \Downarrow \texttt{TYPE\_NUMBER}$

Data (str) $\dfrac{}{(\underline{str},\ s) \Downarrow (\underline{str},\ s)} \Downarrow \texttt{DATA}$

(2) Is_input_available - returns a 1 or 0.

Function returning a 1:

$$\frac{A \quad B}{(\texttt{Is\_input\_available},\ s_1) \Downarrow (\underline{1},\ s_2)} \Downarrow \texttt{IS\_INPUT\_AVAILABLE}_1$$

A is $(\texttt{Is\_input\_available},\ s_1) \Downarrow (\underline{T},\ s_1)$

B is $(\underline{T},\ s_1) \Downarrow (\underline{1},\ s_2)$

Function returning a 0:

$$\frac{A \quad B}{(\texttt{Is\_input\_available},\ s_1) \Downarrow (\underline{0},\ s_2)} \Downarrow \texttt{IS\_INPUT\_AVAILABLE}_2$$

A is $(\texttt{Is\_input\_available},\ s_1) \Downarrow (\underline{F},\ s_1)$

B is $(\underline{F},\ s_1) \Downarrow (\underline{0},\ s_2)$

(3) Is_ok_to_send - returns a 1 or 0.

Function returning a 1:

$$\frac{A \Downarrow B}{(\texttt{Is\_ok\_to\_send},\ s_1) \Downarrow (\underline{1},\ s_2)} \Downarrow \texttt{IS\_OK\_TO\_SEND}_1$$

A is $(\texttt{Is\_ok\_to\_send},\ s_1) \Downarrow (\underline{1},\ s_1)$

B is $(\underline{T},\ s_1)\ (\underline{T},\ s_1) \Downarrow (\underline{1},\ s_2)$

Function returning a 0:

$$\frac{A \Downarrow B}{(\texttt{Is\_ok\_to\_send},\ s_1) \Downarrow (\underline{0},\ s_2)} \Downarrow \texttt{IS\_OK\_TO\_SEND}_1$$

A is $(\texttt{Is\_ok\_to\_send},\ s_1) \Downarrow (\underline{0},\ s_1)$

B is $(\underline{F},\ s_1)\ (\underline{F},\ s_1) \Downarrow (\underline{0},\ s_2)$

(4) Send

$$\frac{(\texttt{Send},\ s_1) \Downarrow (\underline{T},\ s_1)(\underline{T},\ s_1) \Downarrow (\underline{1},\ s_2)}{(\texttt{Send},\ s_1) \Downarrow (\underline{1},\ s_2)} \Downarrow \texttt{SEND}$$

$$\frac{(\texttt{Send},\ s_1) \Downarrow (\underline{F},\ s_1)(\underline{F},\ s_1) \Downarrow (\underline{0},\ s_2)}{(\texttt{Send},\ s_1) \Downarrow (\underline{0},\ s_2)} \Downarrow \texttt{SEND}$$

5. Implementation of the guarded processes:

Let fch_{ij} for $1 \le i \le n$ and $1 \le j \le m$ be a flag to the channel ch_{ij} (where i is the guard number and j is the channel number in that guard, n and m are two positive integers).

Let $(\mathbf{fch_{11}} \wedge \mathbf{fch_{12}} \wedge \ldots \wedge \mathbf{fch_{1m}})$ be FCH_1

Let $(\mathbf{fch_{21}} \wedge \mathbf{fch_{22}} \wedge \ldots \wedge \mathbf{fch_{2m}})$ be FCH_2

Let $(\mathbf{fch_{n1}} \wedge \mathbf{fch_{22}} \wedge \ldots \wedge \mathbf{fch_{nm}})$ be FCH_n

Let $\mathbf{g_1}$, $\mathbf{g_2}$, ….,$\mathbf{g_n}$ be the guard number(integer) for the n guards.

Let **number** be a random integer number.

Let **statement_block_1**, **statement_block_2,** and, **statement_block_n** be the statement blocks for the respective n guarded processes.

Let **if((number ==** $\mathbf{g_i}$**)** $\wedge$ $\mathbf{FCH_i}$**) then statement_block_i** be $\mathbf{G_i}$ for $1 \le i \le n$.

The evaluation relation for the execution of guarded processes in a node which randomly selects and executes a guard and is given as:

Let P be $\underline{number} = \underline{1} + \texttt{(int)rand( ) * } \underline{n}$;

$$FCH_1 \text{ else } FCH_2 \ldots \text{else } FCH_n$$

$$\frac{A \quad B \ \ldots}{(\texttt{while } \underline{true} \texttt{ do P},s_1) \Downarrow (\texttt{while } \underline{true} \texttt{ do P},\ s_i)} \texttt{LOOP}$$

⋮

where i is some integer

A is $(\underline{true},s_1) \Downarrow (\underline{true},s_1)(\texttt{P},s_1) \Downarrow (\texttt{empty},\ s_2)$

B is $(\texttt{while } \underline{true} \texttt{ do P},s_2) \Downarrow (\texttt{while } \underline{true} \texttt{ do P},\ s_3)$

D. Re-write Rules and LAM Codes for the AMPS

In this section, we describe the re-write rules for the AMPS and the respective LAM code necessary for these rules. The re-write rules are defined inductively and they have no premises. The initial and final states of the re-write rules for the AMPS of LIPS is summarised below.

1. **Push** a constant $\underline{n}$ of data type σ (**channel, flag, node_number, vnum, type_number, data**) into the stack S.

$$\boxed{\underline{N}:C} \boxed{S} \boxed{s} \mapsto \boxed{C} \boxed{\underline{n}:\sigma:S} \boxed{s}$$

2. **Fetch** a value from memory location l and place in the stack S.

$$\boxed{\underline{l}:C} \boxed{S} \boxed{s} \mapsto \boxed{C} \boxed{\underline{s(l)}:S} \boxed{s}$$

3. **Assignment instruction** – assign a value P to a memory location of the same data type, i.e, l:=P.

$$\boxed{L:=P:C} \boxed{\underline{c}:S} \boxed{s} \mapsto \boxed{P:ASGNMNT(l):C} \boxed{S} \boxed{s}$$

$$\boxed{ASGNMNT(l):C} \boxed{\underline{n}:S} \boxed{s} \mapsto \boxed{C} \boxed{S} \boxed{s\{l \mapsto \underline{c}\}}$$

4. **`Is_input_available`**

<u>Re-write for the function returning a 1:</u>

`Is_input_available(node_number,vnum):C`	S	S

$\mapsto$

C	S	$s\{l \mapsto \underline{1}\}$

<u>Re-write for the function returning a 0:</u>

`Is_input_available(node_number,vnum):C`	S	S

$\mapsto$

C	S	$s\{l \mapsto \underline{0}\}$

5. **`Is_ok_to_send`**

<u>Re-write for the function returning a 1:</u>

`Is_ok_to_send(node_number, vnum):C`	S	S

$\mapsto$

C	S	$s\{l \mapsto \underline{1}\}$

<u>Re-write for the function returning a 0:</u>

`Is_ok_to_send(node_number,`	S	s

$\mapsto$

C	S	$s\{l \mapsto \underline{0}\}$

6. **`send`** – **Is-ok_to_send and Is_input_available** functions return either a 1 or 0. When a 1 is received, the DS sends the message packet using the **send** function.

`send(data_packet):C`	S	s

$\mapsto$

C	S	$s\{l \mapsto \underline{\texttt{data\_packet}}\}$

Having defined the re-writes, let us compile the LIPS program expressions into LAM codes. Let

$$[\![-]\!] : \texttt{EXP} \rightarrow \texttt{LAMcodes}$$

be the function which takes a LAM program expression and compiles it to LAM code of LIPS. LAM must start in a known state and it is assumed that the program expression fed into the LAM has been type checked by the existing LIPS compiler. There are no explicit rules are needed for the nodes and guarded processes as they can be managed with the existing while construct and alternate construct. The LAM codes for the above re-write rules are as follows:

.

`Is_input_available:`

$$[\![\texttt{is\_input\_available(node\_number, vnum)}]\!] \;\underline{\underline{\texttt{def}}}\; \texttt{IS\_INPUT\_AVAILABLE(node\_number, vnum)}$$

`Is_ok_to_send:`

$$[\![\texttt{is\_ok\_to\_send(node\_number,vnum)}]\!] \;\underline{\underline{\texttt{def}}}\; \texttt{IS\_OK\_TO\_SEND(node\_number,vnum)}$$

`send:`

$$[\![\texttt{send(data\_packet)}]\!] \;\underline{\underline{\texttt{def}}}\; \texttt{SEND(data\_packet)}$$

IV. CONCLUSION

This paper presents an operational semantics for the asynchronous message passing strategy of LIPS. The code needed for the abstract machine that describes the executional behaviour has been included. The communication rules derived not only describe the asynchronous communication that takes place in LIPS but also serve as a reference for implementers of the language. The code added to the LAM for message passing, unambiguously expresses the dynamic behaviour of the message passing in LIPS. The LAM is common (virtually same) to the operational semantics definition of the computational part of LIPS which is not included in this paper.

REFERENCES

[1] S. Bavan, "A language for implementing parallel and distributed systems using asynchronous point-to-point communication", in Nova Science Publishers, Inc. Commak, NY, pp. 109-129, 2001.

[2] S. Bavan and E. Illingworth, "Design and Implementation of Reliable Point-to-Point Asynchronous Message Passing System," in the Proceedings of ICCI, Q8, 2000.

[3] D. Gray, *Introduction to the Formal Design of Real-Time Systems.* Springer Verlag, 1999.

[4] S. Bavan, A. Rajan and G. Abeysinghe, "Asynchronous message passing architecture for a distributed programming language," Submitted for the IADIS Applied Computing 2007.

[5] R. L. Crole, "Operational semantics, abstract machines and correctness," Lecture Notes for the Midlands Graduate School in the Foundations of Computer Science, Leicester, 2006

Magnetization Plateau Enhancement via a Simple Computational Model of Organic Spin Sandwiches

A. Barrañón[1], F. de L. Castillo-Alvarado[2]
[1]Department of Basic Sciences, UAM-Azcapotzalco, Mexico D.F., Mexico
[2]Department of Physics, ESFM-IPN, Mexico D.F., Mexico
Phone (55) 5318-9015 Fax (55) 5318-9540 E-mail: bca@correo.azc.uam.mx

Abstract — **A computer code has been developed that simulates the performance of organic spin sandwiches. Chemical vapor deposition of polymer is simulated by placing monomers in random positions and allowing them to follow a random walk until joining some given linear polymer chain. Spontaneous magnetization in the intermediate region of the sandwich is simulated via Ising model. A genetic algorithm randomly flips a spin whenever the Hamiltonian of the system is optimized in energy space. Magnetization plateau turns out to be maximal when the border spins are parallel and is minimal when border spins are antiparallel. Simulation results are in agreement with recent experimental results for pyrochlore oxide superconductor** KOs_2O_6

Keywords — **Genetic Algorithms, Computer Modelling, Organic Spin Sandwiches, Simulation.**

I. Introduction

Ito et. al. synthesised polyacetilene thin films [1] which came out to be conducting polymers once they were doped [2]. The first organic semiconductor spin-valve was developed by Xiong et. al., based on Alq3 [3], an electron transporter that shows Organic Magnetoresistance. Organic Magnetoresistance is independent of the material characteristics as long as it has been observed in materials with distinct chemical properties such as many organic semiconductors made of different π –conjugated polymers and molecules [4]. This lead to the conjecture that Organic Magnetoresistance should have a simple and general explanation. Nevertheless no model has been able to explain the experimental behavior of organic semiconductors [5] in terms of a simple model. For instance, other authors have recently tried to give a complete description of organic magnetoresistance using theoretical and experimental modelling by means of an excitonic pair modelling based on hyperfine interaction without being successful [6]. Therefore, it is important to develop simplified models of organic semiconductors to contribute to the search of a simple description of organic semiconductor spin valve physical behavior.

In this study we develop a genetic algorithm to simulate a spin valve with a synthetic metal in the middle via a simplified model. This way we obtain computational evidence about magnetization plateau enhancement in spin valves when borders spins are flipped from antiparallel to parallel via a simple model.

II. Methodology

We have built a simplified computational model of a spin sandwich that works as a spin valve in terms of the magnetization enhancement obtained when spins in the ferromagnetic borders turn from antiparallel to parallel.

In the middle of the sandwich linear polymer chains are developed randomly placing monomers that follow a random walk until being absorbed by a given polymer linear chain. Also, spontaneous manetization is supposed to appear in the cavities formed by the linear polymer chains, which is modeled by an Ising Model Hamiltonian.

Spins are successively flipped until system energy is minimized and this way a correlation is obtained between the magnetization and the temperature for distinct values of the non null local conductance probability. In the rest of this section we describe the genetic algorithm used to obtain a spin distribution optimized in energy space, next we describe the genetic algorithm used to build up the linear polymer chains in the middle of the sandwich, and finally we describe the Ising Model Hamiltonian used to model the spontaneous magnetization in the cavities of the linear polymer chains.

1) *Genetic Algorithms*: These kind of algorithms are stochastic global optimization methods based on survivial of the fittest. There is a function f associated to the fitness of the population elements. Given f , the problem in genetic algorithms is to find a set of global maximums of f . A set of points in the space $\{0,1\}^n$ is evolved simulating the adaptation of individuals to the environment. f is a measure of this adaptation. Population for a given time is: $X_n = \{X_n^1, X_n^2, ..., X_n^m\}$ where X_n^i are words of size N. m individuals of a population are selected using a probability distribution that screens the best adapted individuals.

$$\forall h \in \{1,...,m\}, \quad P\left(Y_{n+1}^h\right) = f\left(Y_n^h\right) / \sum_{i=1}^{m} f\left(Y_n^i\right) \tag{1}$$

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 89–91.

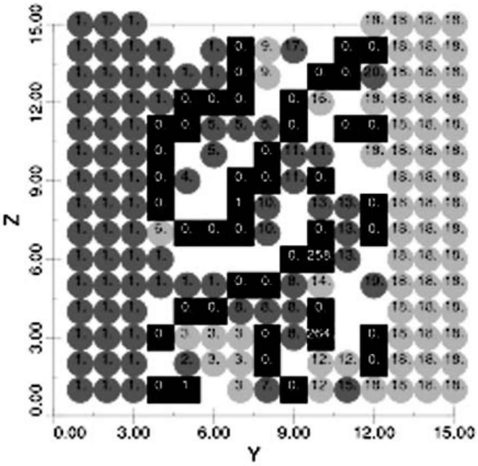


Fig. 1. Shows a sample of an antiparallel ferromagnetic electrodes organic sandwich.

To obtain a new population element of the spin lattice, a site is randomly chosen and its spin is flipped with a probability given by:

$$p = \exp(-\beta H)/[1+\exp(-\beta H]\quad (2)$$

where:

$$\beta = 1/(k_B T)\quad (3)$$

and H is the Hamiltonian of the system.

This is the algorithm employed to simulate the build up of the linear polymer chains and the spontaneous magnetization as explained in the following subsections.

2) *Monte Carlo simulation of linear polymer chains*: In our simulation, a random position in the middle of the sandwich is chosen to place the monomer and afterwards the monomer follows a random walk until it hooks to either the tail of a linear polymer chain or to another monomer, as long as the energy of the system is optimized. This way linear chains are built up in the middle of the sandwich, and in the spaces left between these linear polymer chains, spontaneous magnetization is simulated to obtain the magnetization in the sandwich.

Linear polymer chains have been recently simulated in [7] using Monte Carlo method and simple models where particles interact via a spherically symmetrical repulsive potential obtaining a behavior similar to percolation phenomena and Ising spin systems. Monomers belonging to the polymer chains interact via a screened Coulomb potential obtained by a linearization of the Poisson-Boltzmann equation for the salt:

$$E = (k/2)\sum_i \left|\vec{x}_{i,i+1}\right|^2 + \left[q^2/(4\pi \in_r \in_0)\right]\sum_{i<j}\left[e^{-\bar{\kappa}\left|\bar{x}_{ij}\right|}/\left|\bar{x}_{ij}\right|\right]\quad (4)$$

where $\bar{\kappa}$ is he screening Debye length for a 1:1 salt given by:

$$\bar{\kappa} = q\sqrt{2N_A c_s/(\in_r \in_0 k_B T)}\quad (5)$$

and c_s is salt concentration in molars(M) and N_A is Avogadro number. When $c_s = 0.01M, 0.1M, 1M$, it follows that: $\bar{\kappa} = 0.1992, 0.63, 1.992$ respectively [8].

2) *Spontaneous Magnetization*: Ising model was first used to describe polymer gelation by Flory [9]. An Ising Hamiltonian is used here to obtain an stable configuration of the spin particles once a genetic algorithm is applied:

$$H = -J\sum_{ij} S_i S_j - H\sum_i S_i\quad (6)$$

Magnetization is defined as:

$$M = \left\langle \sum_{i=1}^{N} S_i / N \right\rangle\quad (7)$$

and conductivity between neighbours is given by:

$$\sigma_{ij} = 1 \text{ when } S_i \cdot S_j = 1\quad (8)$$

and; $\sigma_{ij} = 0$, elsewhere. This way a non null local conductance probability is defined as:

$$p(\sigma \neq 0) = \left\langle \sum_{ij}^{N} I(\sigma_{i,j-1} + \sigma_{i,j+1} + \sigma_{i-1,j} + \sigma_{i+1,j} - 1)/N \right\rangle\quad (9)$$

where: $I(x) = 1$, when: $x \geq 0$ and $I(x) = 0$ elsewhere.

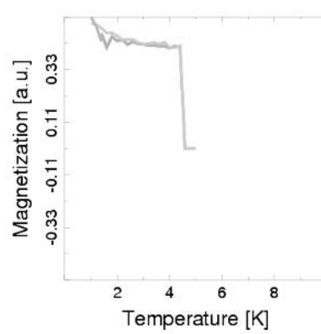


Fig. 2. Shows enhanced magnetization for the parallel arrrangement. Magnetization as a function of temperature for the organic spin-valve simplified model with parallel ferromagnetic borders. A critical temperature close to 5K is observed, as well as a magnetization plateau of about 0.88.

This page intentionally blank

III. Results

When the ferromagnetic border spins are antiparallel (Fig. 1), magnetization is minimal attaining a value of about 0.02, while in the parallel ferromagnet electrodes arrangement, spontaneous magnetization turns out to be maximal attaining a value of about 0.88 (Fig. 2), for high values of the non null local conductance probability.

Therefore magnetization changes in about 4300% and hence this system works as a valve that is magnetized when the electrodes are spin parallel and is scarcely magnetized when the electrodes are spin antiparallel. In both cases, magnetization plateaus correspond to high values of the probability of no null local conduction. This is in qualitative agreement with a experimental study of pyrochlore oxide superconductor KOs_2O_6 [10].

IV. Conclusion

We have obtained a simplified model based on a genetic algorithm that replicates the behavior of spin valves in terms of magnetic plateau enhancement due to parallel arrangement of the ferromagnetic electrodes. This contributes to the search for a simple model of organic spin valves.

References

[1] T. Ito, H. Shirakawa, and S. Ikeda, “Simultaneous polymerization and formation of polyacetylene film on the surface of concentrated soluble ziegler type catalyst solution”, *J. Polym. Sc., Polym. Chem. Ed.*, vol. 12, p. 11, 1974.

[2] C. K. Chiang, C.R. Fincher, Jr., Y.W. Park, J. Heeger, H. Shirakawa, E.J. Louis et al, “Electrical Conductivity in Doped Polyacetylene”, *Phys. Rev. Lett.,* vol. 39, p. 1098, 1977.

[3] Z. H. Xiong, D. Wu, Z. V. Vardeny, and J. Shi, “Giant magnetoresistance in organic spin-valves”, *Nature,* no. 427, p. 821, 2004.

[4] Ö Mermer, G. Veeraraghavan, and T.L. Francis M, “Large magnetoresistance at room-temperature in small molecular weight organic semiconductor sandwich devices”, *condmat/0501124*, 2005.

[5] Ö Mermer, G. Veeraraghavan, T.L. Francis, Y. Sheng, D.T. Nguyen, M. Wohlgennant et. al., “Large magnetoresistance in nonmagnetic π -conjugated semiconductor thin film devices”, *Phys. Rev. B,* vol. 72, p. 205202, 2005.

[6] Y. Sheng, D. T. Nguyen, G. Veeraraghavan, Ö. Mermer, M. Wohlgenannt and U. Scherf, “Hyperfine interaction and magnetoresistance in organic semiconductors”, *cond-mat/0602282,* 2006.

[7] Y. Norizoe and T. Kawakatsu, “Monte Carlo simulation of string-like colloidal assembly”, *Europhys. Lett.*, vol. 72, no. 4, pp. 583-589, 2005.

[8] B. Jönson, C. Peterson and B. Söderberg, “A variational Approach to the Structure and Thermodynamics of Linear Polyelectrolytes with Coulomb and Screened Coulomb Interactions”, *Journal of Physical Chemistry,* no. 99, p. 1251, 1995.

[9] P. J. Flory, “Molecular size distribution in three dimensional polymers. I. Gelation”, *Journal of the American Chemical Society*, vol. 63, pp. 3083-3090, 1941.

[10] J.I. Yamaura, S. Yonezawa, Y. Muraoka, Z. Hiroi, "Crystal structure of the pyrochlore oxide superconductor KOs2O6", *arxiv:/cond-mat/0512683*, 2005.

Affine Invariant Feature Extraction Using a Combination of Radon and Wavelet Transforms

Asad Ali[1] S.A.M Gilani[1] Umer Shafique[2]
1. Faculty of Computer Science and Engineering, 2. Faculty of Electronic Engineering
GIK Institute of Engineering Sciences and Technology
Topi-23460, Swabi, Pakistan
{ aali, asif }@giki.edu.pk, umar.shafique@ieee.org

Abstract - **The paper presents a new framework for the extraction of region based affine invariant features with the view of object recognition in cluttered environments using the radon transform. The presented technique first normalizes an input image by performing data pre-whitening which reduces the problem by removing shearing deformations. Then four invariants are constructed by exploiting the properties of radon transform in combination with wavelets which enable the analysis of objects at multiple resolutions. The proposed technique makes use of an adaptive thresholding technique for the construction of invariants. Experimental results conducted using three different standard datasets confirm the validity of the proposed approach. Beside this the error rates obtained in terms of invariant stability in noisy conditions are significantly lower when compared to the method of moment invariants and the proposed invariants exhibit good feature disparity.**

Keywords - **Affine invariants, Radon transform, Feature Extraction, Geometric transformations, Wavelets, Pattern recognition.**

I. INTRODUCTION

Recognizing objects when subjected to perspective transformations in noisy and cluttered environments is one of the primary tasks in computer vision. Viewpoint related changes of objects can broadly be represented by weak perspective transformation which occurs when the depth of an object along the line of sight is small compared to the viewing distance [10]. This reduces the problem of perspective transformation to the affine transformation which is linear.

Constructing invariants to certain groups Euclidean, affine and projective deformations hold potential for wide spread applications for industrial part recognition [11], handwritten character recognition [12], identification of aircrafts [13], and shape analysis [14] to name a few.

The affine group includes the four basic forms of geometric distortions, under weak perspective projection assumption, namely translation rotation, scaling and shearing. Finding a set of descriptors that can resist geometric attacks on the object contour can act as a good starting point for the more difficult projective group of transformations.

In this paper we propose a new method of constructing invariants which is based on normalizing an affine distorted object using data pre-whitening which removes shearing distortion from the object on the Cartesian grid. Then a set of four invariants are constructed using a combination of radon and wavelet transform over the region of support of the object and using the un-parameterized object boundary.

The rest of the paper is organized as follows. In section 2 we review some the previously published works, section 3 describes the proposed method in detail and section 4 provides experimental results and comparisons with previously published techniques. But before we move ahead let us have a brief overview of radon transform.

A. Radon Transform

Introduced by Peter Toft [8] and related to Hough transform, it has received much attention in the past couple of years with applications emphasizing its use for line detection and localization [9]. Primarily it is able to transform two dimensional images with lines into a domain of possible line parameters where each line gives a peak at the corresponding orientation.

The two dimensional discrete radon transform for an images *f(x,y)* can be expressed as:

$$R(\rho,\theta) = \iint f(x,y)\delta(\rho - x\cos\theta - y\sin\theta) \quad (1)$$

where ρ is the distance of the line from the origin, δ is the dirac delta and θ is the orientation. Radon transform satisfies linearity, scaling, rotation and skewing which relates it directly to the affine group of transformations.

The above properties combined with the capability of radon transform to detect lines under high noise levels was the primary motivation for its selection as a tool for invariant feature extraction.

II. RELATED WORK

Importance of constructing invariants to certain geometric transformations can be gauged from the fact that research has been conducted by many during the last two decades which can broadly be classified into two groups namely: Region based and Contour based invariant descriptors.

Region based techniques can further be classified into two groups: symmetric and asymmetric. In the context below we review some of the region based techniques that are most related to the present work.

Hu [1] introduced a set of seven affine moment invariants which were later corrected in [2] and have widely been used by the pattern recognition community. They are

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 93–97.

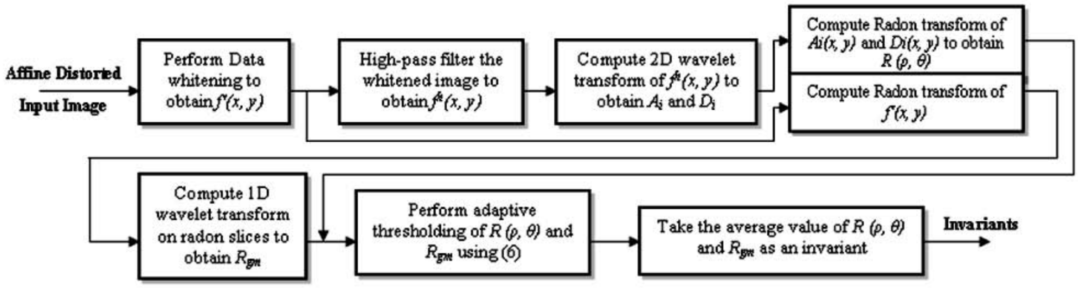


Fig. 1 shows the complete system diagram for the construction of region based invariant descriptors.

computationally simple and invariant to translation, scaling, rotation and skewing but suffer from several drawbacks like: information redundancy, which occurs because the basis used in their construction are not orthogonal, noise sensitivity, higher order moments are very sensitive to noise and illumination changes, finally large variation in the dynamic range of values may cause numerical instability with larger object size.

Some of the problems associated with moment invariants were addressed by Teague [3] who proposed the use of continuous orthogonal moments with higher expressive power. Zernike and Legendre moments were introduced by him based on the zernike and legendre polynomials. Zernike moments have proven to better represent object features besides providing rotational invariance and robustness to noise and minor shape distortions. But several problems are associated with the computation of zernike moments like numerical approximation of continuous integrals with discrete summations which leads to numerical errors affecting the properties such as rotational invariance and increase in computational complexity when the order of the polynomial becomes large.

Zhang et al [4] solved the problem of noise sensitivity associated with the framework of moment invariants and constructed invariants using the framework proposed in [2] in the spatial domain after Fourier filtering. An adaptive thresholding technique was developed as part of the framework to enable the establishment of the correspondence between the affine related images under high noise levels. But the technique has only been shown to work for symmetric images suffering from RST group of distortions.

More recently Petrou et al [5][6] introduced the trace transform for affine invariant feature extraction. Related to integral geometry and similar to radon transform however more general than either of them it computes image features along line integrals and performs calculations of any functional over the group of transformations in a global manner. They have used a set of three functionals namely line, diametrical and circus for the computation of invariant features. The major drawback is the computational cost which increases exponentially with the number of trace functionals.

Finally Heikkila et al [7] introduced the concept of autoconvolution across multiple scales of an input image and constructed a set of 29 invariants in the Fourier domain. They use the expected value of autoconvolution as an invariant. Although their technique produces excellent results but their method results in feature overlapping across different frequencies which serves as a major limitation in object classification.

In an attempt to solve the problems mentioned above, the present work makes use of the radon transform in combination with wavelets to construct a set of invariants that can be used for recognizing objects under the affine transformations coupled with high noise distortion levels.

In short we improve on many of the short comings mentioned above.

III. Proposed technique

We propose a three step process for the construction of region based invariant descriptors of the objects. The first step acts as foundation for second and third steps in which radon transform is applied and then invariants are constructed. In the context below we provide the detailed description of each step.

A. Data Pre-whitening

Let us consider an image *f(x,y)* which is parameterized as Y(t) = **[x(t), y(t)]** with parameter *t* on a plane by performing raster scanning on the coordinate axis. Thus a point from *Y(t)* under an affine transformation can be expressed as:

$$\begin{aligned} \tilde{x}(t) &= a_0 + a_1 x(t) + a_2 y(t) \\ \tilde{y}(t) &= b_0 + b_1 x(t) + b_2 y(t) \end{aligned} \tag{2}$$

The above equations can be written in matrix form as:

$$\begin{aligned} \begin{bmatrix} \tilde{x}(t') \\ \tilde{y}(t') \end{bmatrix} &= \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} + \begin{bmatrix} a_0 \\ b_0 \end{bmatrix} \\ \begin{bmatrix} \tilde{x}(t') \\ \tilde{y}(t') \end{bmatrix} &= P \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} + B \\ Y'(t') &= PY(t) + B \end{aligned} \tag{3}$$

Next whitening is performed on *Y'(t')* by computing the Eigen value decomposition of covariance matrix as:

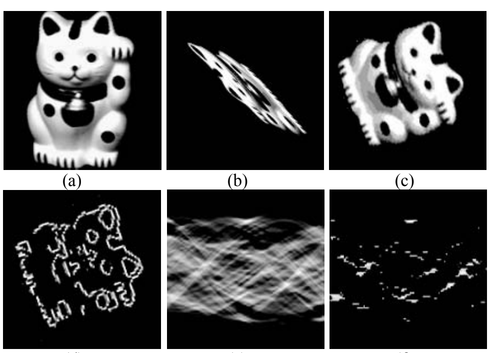
$$Y' Y'^{T} = EDE^{T}$$
$$\tilde{Y}' = ED^{-1/2}E^{T}Y' \quad (4)$$

where E is the orthogonal matrix of eigenvectors of $\{ Y' Y'^{T} \}$ and D is the diagonal matrix of eigen values. As a result the data Y' becomes uncorrelated after this step which effectively removes shearing distortion from the input image. Let us call the obtained image $f'(x, y)$.

B. Invariant Wavelet-Radon Descriptors

Invariant descriptors of the image can now be computed by following the steps mentioned below:

1. Perform high pass filtering on $f'(x, y)$ to obtain image $f^{h}(x, y)$.
2. Compute the 2-Dimensional wavelet transform of $f^{h}(x, y)$ with decomposition at level i to obtain $A_i(x,y)$ and $D_i(x,y)$, where $A_i(x,y)$, $D_i(x,y)$ represent the approximation and detail coefficients.
3. Project the output $A_i(x,y)$ and the absolute additive of $D_i(x,y)$ across the horizontal, vertical and diagonal directions onto different orientation slices using (1) to obtain the radon transform coefficients $R(\rho,\theta)$. The orientation angles of the slices are ordered in counter clockwise direction with an angular step of Π/n, where n is the number of orientations.
4. Estimate: $\Delta = \max(R) * K$, (5)
   where K is a predefined constant between {0-1}.
5. Perform thresholding of $R(\rho,\theta)$ as:
   $$Q = \begin{cases} R(\rho,\theta) & \text{if} \quad R(\rho,\theta) > \Delta \\ 0 & \text{otherwise} \end{cases} \quad (6)$$
6. Compute average value of Q which is an invariant.
7. Repeat from step 2 with decomposition at level j, where $i < j$.



(a) (b) (c) (d) (e) (f)

Fig. 2 (a) Original Image. (b) Affine transformed input image. (c) Image obtained after data whitening. (d) Approximation coefficients of the wavelet transform. (e) Corresponding Radon transform output. (f) Output obtained after adaptive thresholding.

Invariant I_1 and I_2 reported in section IV are constructed using the above methodology, where as Fig. 2 provides a brief overview of the steps mentioned above.

C. Invariant Ridgelet Descriptors

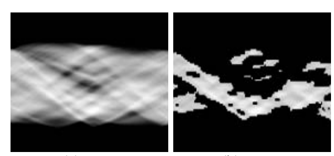
Here we propose another technique for the construction of invariant descriptors using the ridgelet transform. For each $a > 0$, each $b \in R$ and each $\theta \in [0, 2\Pi]$ the bivariate ridgelet $\psi_{a,b,\theta} : R^2 \to R$ is defined as:

$$\psi_{a,b,\theta} = a^{-1/2}\psi((x_1 \cos\theta + x_2 \sin\theta - b)/a) \quad (7)$$

where $\psi(.)$ is a wavelet function. A ridgelet is nothing but a constant along the line $x_1 \cos\theta + x_2 \sin\theta$, equation (7) shows that a ridgelet can be represented in terms of the radon and wavelet transforms which simplifies the construction of invariants using the ridgelets. The stepwise process for the construction of invariant descriptors is detailed below:

1. Project $f'(x,y)$ onto different orientation slices to obtain the radon transform coefficients $R(\rho,\theta)$. The orientation slices pass though the rectangular image with a lag of ρ between each slices.
2. Compute one dimensional wavelet transform on the orientation slices $R(\rho,\theta)$ with decomposition at level i and j to obtain the ridgelet coefficients R_{gi} and R_{gj}.
3. Estimate the threshold Δ from equation (5) with R_{gm} as input, where $m \in \{i, j\}$.
4. Perform thresholding of R_{gm} as per equation (6) to obtain R_{gm}'.
5. Take average value of R_{gm}' as an invariant.
6. Repeat from step 2 with decomposition at level k, where $i < j < k$.

Invariant I_3 and I_4 reported in section IV are constructed using the above methodology where as Fig. 3 provides a brief overview of the steps mentioned above. Fig. 1 shows the complete system diagram and elaborates the above mentioned operations in a sequential and precise manner.



(a) (b)

Fig. 3(a) Ridgelet coefficients obtained for image in figure 2(c). (b) Corresponding output obtained after adaptive thresholding of the coefficients.

IV. EXPERIMENTAL RESULTS

The proposed technique was tested on a 2.4 GHz Pentium 4 machine with Windows XP and Matlab as the development tool. The datasets used in the experiments include the Coil-20 dataset, MPEG-7 Shape-B datasets and English alphabets dataset. Coefficients at level two are used in the construction of invariant $\{I_1, I_2, I_3, I_4\}$ where as decomposition is performed using the daubechies wavelets and the value of K and n used is 0.70 and 180.

This section is divided into three parts first we demonstrate the stability of the four invariants against five different affine transformations then we demonstrate the feature discrimination capability of the invariants and finally we provide a comparative analysis of the invariants with the method of moment invariants [2].

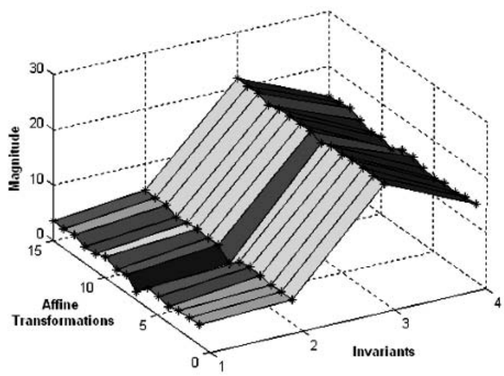


Fig. 4 shows the 3D surface plot of the four invariant descriptors for the object in figure 2(a) against fifteen affine transformations Order of invariants has been changed for clarity.

Table 1 shows the magnitude of the invariant descriptors $\{I_1, I_2, I_3, I_4\}$ against different affine transformations for the object shown in figure 2(a). In the table following notation is used: Rotation (R) in degrees, Scaling (S) along x and y axis, Shear (Sh) along x and y axis, Translation (T) and mirror (M). The figures in brackets represent the parameters of the transformation.

To further elaborate and demonstrate invariant stability fig. 4 shows the 3D surface plot of four invariants against fifteen affine deformations covering all aspects of the affine group in a precise manner.

TABLE 1
SHOWS THE MAGNITUDE OF THE INVARIANTS AFTER APPLYING DIFFERENT AFFINE TRANSFORMATIONS.

Transformation	Object 4 from coil-20 dataset, figure [2(a)]			
	I_1	I_2	I_3	I_4
Original Image	13.97	21.33	4.32	3.54
R(40), S(3,1)	13.72	21.95	3.94	2.42
R(135), S(2,3), T, M	13.45	21.36	4.63	2.92
R(45),Sh(2.05,1.0),T	13.96	21.47	4.61	2.89
R(165), S(3,3), Sh(1,2), T, M	14.51	21.54	4.86	3.14
R(230), S(4,1), Sh(3,3), T, M	13.37	21.88	4.37	2.73

Fig. 5 demonstrates the feature discriminating capability of the proposed invariants for different objects from the coil-20 dataset. A classifier can be trained which makes use of the proposed set of invariants to perform object recognition.

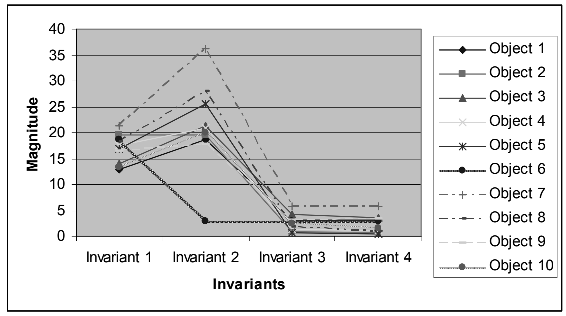


Fig. 5 shows the feature discrimination capability of the proposed invariants for ten different objects from the coil-20 dataset.

Finally fig. 6 compares the proposed invariants with the method of moment invariants [2] (first six invariants are used) at different noise (salt & pepper) variance levels. A set of 20 affine transformations are used in the experiment. The results are averaged over the coil-20 dataset. The metric used for computing the error is σ / μ. The error has been averaged for all invariants.

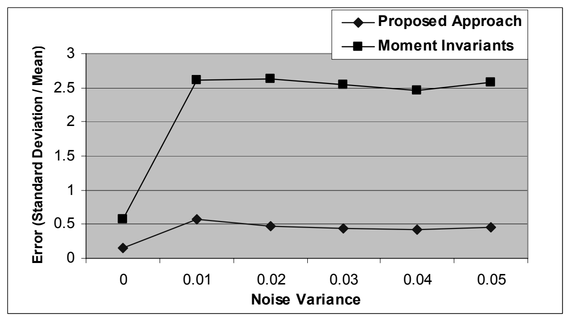


Fig. 6 shows the comparison between the proposed set of invariants and the method of moment invariants [2] against different noise variance levels.

Obtained results show significant reduction in error thus validating the proposed framework.

CONCLUSION

In this paper we have presented a new framework for the construction of affine invariant descriptors using the radon and wavelet transforms. Experimental results demonstrate the robustness of invariants against different affine transformations and under various noise levels, which only became possible through the use of radon transform. Beside this the use of wavelet transform provided the much needed discriminative power to the proposed set of invariants. Presently, work is in progress to extend the framework to handle the projective group of transformations and estimation of the affine parameters, in future we intend to build an intelligent classifier for performing object recognition over a large dataset based on the proposed invariants.

This page intentionally blank

ACKNOWLEDGMENT

The authors would like to thank National Engineering & Scientific Commission (NESCOM) for their financial support, GIK Institute of Engineering Sciences & Technology for facilitating this research and University of Columbia for providing the Coil-20 dataset.

REFERENCES

[1] M.K.Hu, "Visual Pattern recognition by moment invariants", IRE Transactions on information theory, 1962.
[2] J.Flusser, T. Suk, "Pattern recognition by affine moment invariants", Pattern recognition, vol.26, no.1, 1993.
[3] M.Teague, "Invariant image analysis using the general theory of moments", Journal of Optical society of America, vol. 70, no.8, August 1980.
[4] Y. Zhang, Y. Zhang, C. Wen, "Recognition of Symmetrical Images Using Affine Moment Invariants in both Frequency and Spatial Domains", Pattern Analysis & Applications, vol 5, no. 3, 2004.
[5] M.Petrou, A.Kadyrov, "Affine invariant features from the trace transform", IEEE transactions on pattern analysis and machine intelligence, vol.26, no.1, January 2004.
[6] A. Kadyrov, M.Petrou, "The Trace transform and its applications", IEEE Transactions on pattern analysis and machine intelligence, vol.23, no.8, August 2001.
[7] E.Rahtu, M.Salo, J.Heikkila, "Affine invariant pattern recognition using multiscale autoconvolution", IEEE Transactions on pattern analysis and machine intelligence, vol.27, no.6, June 2005.
[8] P.Toft, "The Radon Transform: Theory and Implementation", Ph.D. thesis, Department of Mathematical Modelling, Technical University of Denmark, June 1996.
[9] P.Toft, "Using the generalized radon transform for the detection of curves in noisy images", IEEE International conference on acoustics, speech and signal processing, 1996.
[10] J.Mundy, A.Zisserman, "Geometric invariance in computer vision", MIT Press, Cambridge, MA.
[11] Y.Lamdan, J.T.Schwartz, "Affine Invariant Model based object recognition", IEEE Transactions on robotics and automation, vol.6, no.5, October 1990.
[12] T. Wakahara, K.Adaka, "Adaptive Normalization of handwritten characters using global-local affine transformations", IEEE Transactions on pattern analysis and machine intelligence, vol.20, no.12, December 1998.
[13] M.I.Khalil, M.Bayoumi, "Affine invariants for object recognition using the wavelet transform", Pattern recognition letters, no.23, 2002.
[14] I.E.Rube, M.Ahmed, M.Kamel, "Coarse to fine multiscale affine invariant shape matching and classification", Proc of 17th International Conference on Pattern recognition, 2004.
[15] S.Mallat, "A Wavelet Tour of Signal Processing" 2nd Edition, Academic Press, 1999.
[16] S.Manay, D.Cremers, B.Hong, A.Yezzi, S.Soatto, "Integral Invariants for shape Matching", IEEE Transactions on pattern analysis and machine intelligence, vol. 28, no. 10, October 2006.
[17] F.Colonna, G.Easley, Generalized discrete radon transforms and their use in the ridgelet transform, Journal of Mathematical imaging and vision vol 23, 2005.

Evaluating the Predictability of Financial Time Series A Case Study on Sensex Data

P. Bagavathi Sivakumar [1] , Dr. V. P. Mohandas [2]
[1] Assistant Professor, Department of Computer Science and Engineering,
[2] Professor and Chairman, Department of Electronics and Communication Engineering,
School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore – 641 105, India.

Abstract:

A discrete- time signal or time series is set of observations taken sequentially in time, space, or some other independent variable. Examples occur in various areas including engineering, natural sciences, economics, social sciences and medicine. Financial time series in particular are very difficult to model and predict, because of their inherent nature. Hence, it becomes essential to study the properties of signal and to develop quantitative techniques. The key characteristics of a time series are that the observations are ordered in time and that adjacent observations are related or dependent. In this paper a case study has been performed on the BSE and NSE index data and methods to classify the signals as Deterministic, Random or Stochastic and White Noise are explored. This pre-analysis of the signal forms the basis for further modeling and prediction of the time series.

Keywords:
Time Series, Signal Analysis, Time Series Analysis, Deterministic, Stochastic.

I. INTRODUCTION

A discrete-time signal or time series [1] is a set of observations taken sequentially in time, space or some other independent variable. Many sets of data appear as time series: a monthly sequence of the quantity of goods shipped from a factory, a weekly series of the number of road accidents, hourly observations made on the yield of a chemical process and so on. Examples of time series abound in such fields as economics, business, engineering, natural sciences, medicine and social sciences.

An intrinsic feature of a time series is that, typically, adjacent observations are related or dependent. The nature of this dependence among observations of a time series is of considerable practical interest. Time Series Analysis is concerned with techniques for the analysis of this dependence [2]. This requires the development of models for time series data and the use of such models in important areas of application.

A discrete-time signal x (n) is basically a sequence of real or complex numbers called samples. Discrete-time signals can arise in various ways. Very often, a discrete-time signal is obtained by periodically sampling a continuous-time signal, that is $x(n) = x_c(nT)$, where $T = 1 / F_s$ is the sampling period and F_s is the sampling frequency. At other times, the samples of a discrete-time signal are obtained by accumulating some quantity over equal intervals of time, for example, the number of cars per day traveling on a certain road. Financial signals, like daily stock market prices are inherently discrete-time.

When successive observations of the series are dependent, the past observations may be used to predict future values. If the prediction is exact, the series is said to be deterministic. We cannot predict a time series exactly in most practical situations. Such time series are called random or stochastic, and the degree of their predictability is determined by the dependence between consecutive observations. The ultimate case of randomness occurs when every sample of a random signal is independent of all other samples. Such a signal, which is completely unpredictable, is known as White noise and is used as a building block to simulate random signals with different types of dependence. To properly model and predict a time series, it becomes important to fundamentally and thoroughly analyze the signal it self, and hence there is a strong need for signal analysis.

II. SIGNAL ANALYSIS

The classification of signals as deterministic, random or stochastic and white noise is very important in deciding about models and methods for prediction. The signal analysis has to be viewed in this regard. The primary of goal of signal analysis is to extract useful information that can be used to understand the signal generation process or extract features that can be used for signal classification purposes. Typical applications of signal analysis include detection and classification of radar and sonar targets, speech and speaker recognition, detection and classification of natural and artificial seismic events, event detection and classification in biological and financial signals, efficient signal representation for data compression, image processing, etc.

A. *Signal analysis Techniques*

The main objective of signal analysis is the development of quantitative techniques to study the properties of a signal and the differences and similarities between two or more signals from the same or different sources. The major areas of random signal analysis are:

1. Statistical analysis of signal amplitude, that is the sample values

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 99–104.

2. Analysis and modeling of the correlation among the samples of an individual and
3. Joint signal analysis that is, simultaneous analysis of two signals in order to investigate their interaction or interrelationships.

The various random signal analysis techniques found in the literature are shown in the Figure 1.

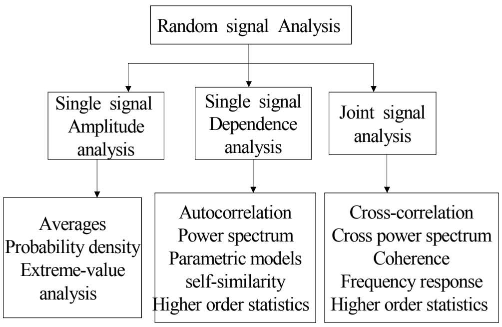


Fig. 1. **Signal analysis Techniques.**

The prominent tool in signal analysis is spectral estimation, which is a generic term for a multitude of techniques used to estimate the distribution of energy or power of a signal from a set of observations. Spectral estimation finds many applications in areas such as medical diagnosis, speech analysis, seismology and geophysics, nondestructive fault detection, testing of physical theories and evaluating the predictability of time series.

The range of values taken by the samples of a signal and how often the signal assumes these values together determine the signal variability. The signal variability can be seen by plotting the time series and is quantified by the histogram of the signal samples, which shows the percentage of the signal amplitude values with in a certain range. The numerical description of signal variability, which depends only on the value of the signal samples and not on their ordering, involves quantities such as mean value, median, variance and dynamic range.

B. Correlation

Scatter plots give the existence of correlation, to obtain quantitative information about the correlation structure of a time series x (n) with zero mean value.

The better estimate is to use the empirical normalized autocorrelation sequence, which is an estimate of the theoretical normalized autocorrelation sequence. For lag L = 0, the sequence is perfectly correlated with itself and we get the maximum value of 1. If the sequence does not change significantly from sample to sample, the correlation of the sequence with its shifted copies, though diminished, is still close to 1. Usually the correlation decreases as the lag increases because distant samples become less and less dependent. Signals, whose empirical autocorrelation decays fast, such as exponential, have short memory or short-range dependence. If the empirical autocorrelation decays very slowly, as a hyperbolic function does, then the signal has a long memory or long-range dependence.

The spectral density function shows the distribution of signal power or energy as a function of frequency. The autocorrelation and the spectral density of a signal form a Fourier transform pair and hence contain the same information. However, they present this information in different forms, and one can reveal information that cannot be easily extracted from the other.

Various tools found in the literature [3][4][5][6] which are used in the classification of signals are:

- Power spectrum
- Correlation dimension
- Embedding parameters
- Lyapunov exponent
- Surrogate data study
- Deterministic Versus Stochastic plot
- Structure function
- Average Mutual Information
- False Nearest Neighbors
- Embedding dimension
- Hurst exponent
- Recurrence Histogram
- Spatio-Temporal Entropy

III. EXPERIMENTAL RESULTS

The data taken (1447 samples) for this experimentation are

- BSE index for the years 1999 – 2004
- NSE index for the years 1999 – 2004

A. Time Series Chart

Time series chart is the regular 2-dimensional graph that shows the dynamics of the scalar series in time. Time series chart are shown in Figure 2 and 3.

B. Auto Correlation Function

The auto correlation functions for BSE and NSE index are shown in Figure 4 and 5. This shows long range dependence (LRD) for both NSE and BSE.

C. Correlation between NSE and BSE

Pearson coefficient r, gives a measure of association between two variables.

r = 1 perfect Positive relationship
r = 0 No relationship
r = -1 perfect negative relationship

Relationship between NSE and BSE for the period is explored and the results are shown in Table I.

A total of 1447 sample data was taken for the experiment. The correlation is significant at the 0.01 level (2-tailed).

D. Average Mutual Information

Mutual information function can be used to determine the “optimal” value of the time delay for the state space reconstruction, as first proposed in an article by Andrew M. Fraser and Harry L. Swinney in "Independent coordinates for strange attractors from mutual information", Phys. Rev. A 33 (1986) pp. 1134-1140. The idea is that a good choice for the time delay T is one that, given the state of the system X(t), provides new information with measurement at X(t+T).

Mutual information is the answer to the question, "Given a measurement of X(t), how many bits on the average can be predicted about X(t+T)?" A graph of I(T) starts off very high (given a measurement X(t), we know as many bits as possible about X(t+0)=X(t)). As T is increased, I(T) decreases, then usually rises again. It is suggested that the value of time delay where I(T) reaches its first minimum be used for the state space reconstruction.

Average mutual information chart for BSE and NSE index are shown in Figure 6 and 7.

E. Correlation Dimension:

For Chaotic signals the Correlation Dimension will saturate as Embedding dimension is increased, and for Stochastic signals the Correlation Dimension will be equal to embedding dimension. The charts are shown in Figure 8 and 9.

BSE Index appears to be more random than NSE for the period 1999 to 2004. Due to this if we choose to model the index as having contributions for deterministic and stochastic models, BSE index will have more contributions for stochastic model than that for NSE.

F. Recurrence Histogram

The recurrence histogram shows the characteristic periodicity in time series. The recurrence histograms are shown in Figure 10 and 11.

G. Spatio-Temporal Entropy

Spatio-Temporal Entropy measures the image "structureness" in both space and time domains. The following range of spatio-temporal entropy should be expected for different signals:

Signal	Spatio-Temporal Entropy
---------	-----------------------------
Periodic	near 0%
Chaotic	0 .. 100%
Random	near 100%

For NSE and BSE the values are found as

NSE: 55 %

BSE: 51 %

H. Applications

As discussed above these techniques are very useful in the analysis of signals and in modeling them. The important areas and applications are

- Population growth (Plants, Animals)
- Meteorological data (Temperature, etc.)
- El Nino (Pacific ocean temperature)
- Seismic waves (Earthquakes)
- Tidal levels
- Astrophysical (sunspots, Cephids, etc.)
- Fluid fluctuations / turbulencePhysiological (EEG, EGC, etc.)
- Epidemiological (Diseases)
- Music and Speech
- Geological core samples
- Written text - sequence of ASCII codes
- Lorenz attractor – x(t) sampled at regular intervals for flow

Visual Recurrence Analysis, version 4.7 software program was used for some experimentation.

IV. Conclusion

The importance of signal analysis or pre-analysis of signals in the modeling and prediction of time series was discussed in this paper. The signal analysis plays a vital and an important role in evaluating the predictability of a time series. The various methods and techniques of signal analysis are applied for the financial time series namely BSE and NSE index and the results are shown and discussed.

References

[1] Dimitris G. Manolakis, Vinay K. Ingle, Stephen M. Kogon, *Statistical and Adaptive Signal Processing, Spectral Estimation, Signal Modeling, Adaptive Filtering and Array Processing* (McGraw-Hill International Editions, 2000)

[2] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, Time Series Analysis Forecasting and Control (Pearson Education, Inc. 2004)

[3] J. C. Parikh, *Stochastic Processes and Financial Markets* (Narosa Publishing House, 2003)

[4] Stephen A. Delurgio, *Forecasting Principles and Applications,* (McGraw-Hill International Editions, 1998)

[5] Holger Kantz, Thomas Schreiber, *Nonlinear time series analysis* (Cambridge University Press, 1996)

[6] Yuh-Dauh- Lyuu, *Financial Engineering and Computation, Principles, Mathematics, Algorithms* (Cambridge University Press, 2002)

[7] Richard O.Duda, Peter E. Hart, David G. Stork, *Pattern Classification* (A Wiley-Inter science Punlication, John Wiley & Sons, Inc. 2000)

[8] Andreas S. Andreou, Testing the predictability of the Cyprus Exchange: The case of an emerging market, proceedings of the IEEE –INNS – ENNS *International Joint Conference on Neural Networks* (IJCNN’00), 2000.

[9] Ashok Razdan, Scaling in the Bombay stock Exchange index, *PRAMANA- journal of Physics, Vol. 58. No. 3, March 2002, pp 537-544.*

[10] Sameer Singh, Elizabeth stuart, A Pattern matching tool for Time-Sereis Forecasting.

Fig. 2. Time series chart for BSE - 1999 to 2004

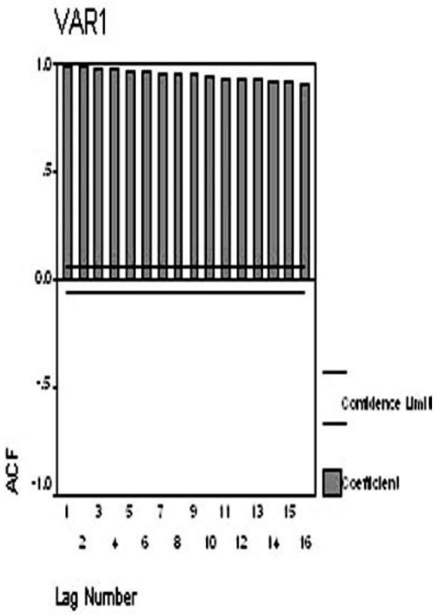


Fig. 4. Auto Correlation Chart for BSE

Fig. 3. Time series chart for NSE – 1999 to 2004

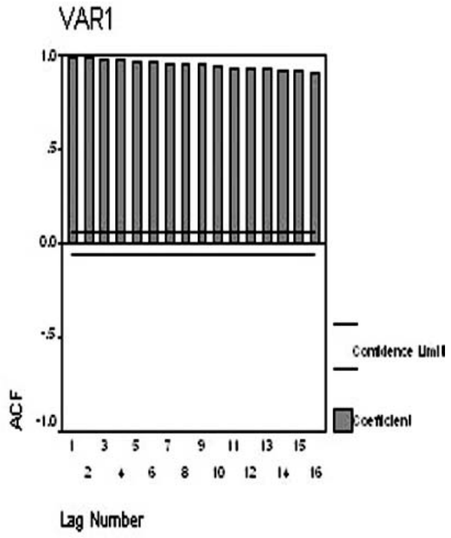


Fig. 5. Auto Correlation Chart for NSE

Table I. Correlation coefficient for BSE and NSE

Correlations

		BSE	NSE
Pearson Correlation	BSE	1.000	.960**
	NSE	.960**	1.000
Sig. (2-tailed)	BSE	.	.000
	NSE	.000	.
N	BSE	1447	1447
	NSE	1447	1447

**. Correlation is significant at the 0.01 level (2-tailed).

N – Number of samples: 1447.

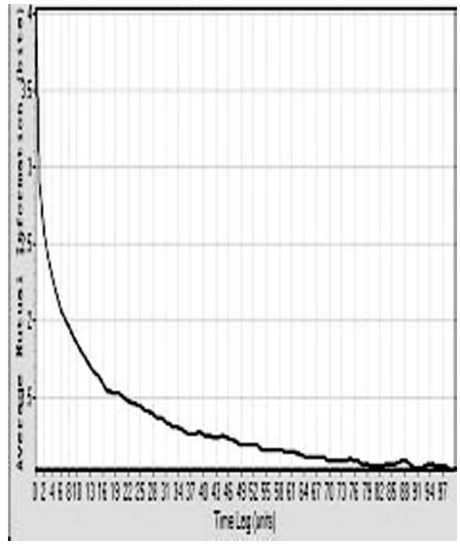


Fig. 7. AMI chart for NSE: Time lag 29

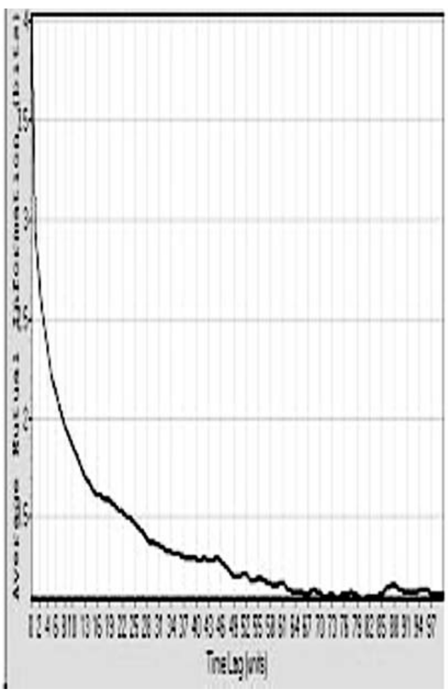


Fig. 6. AMI chart for BSE: Time lag 41

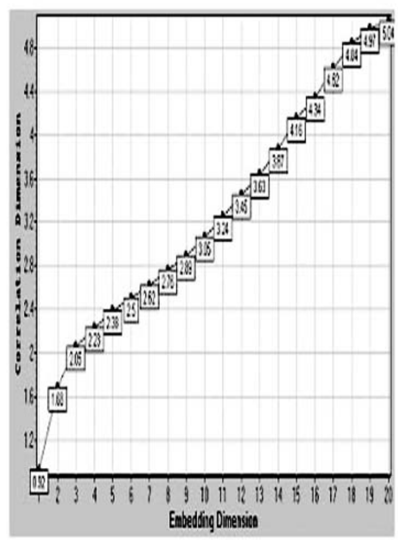


Fig. 8. BSE Correlation Dimension

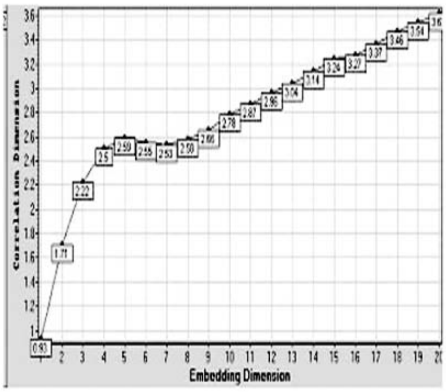


Fig. 9. NSE Correlation Dimension

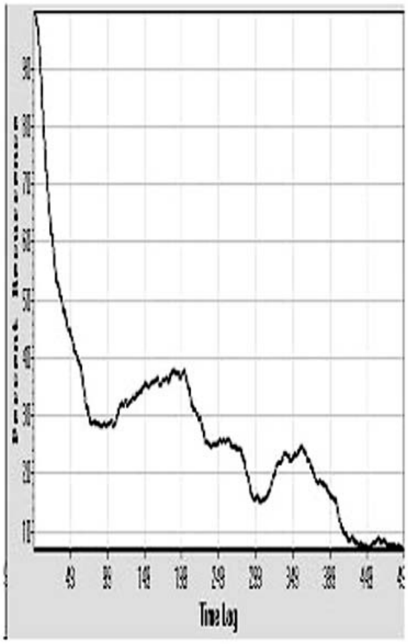


Fig. 11. BSE Recurrence Histogram

Fig. 10. NSE Recurrence Histogram

Activity-Based Software Estimation using Work Break down Structure

Basavaraj M.J
Perot Systems, EPIP Phase II, Whitefield Industrial Area,
Bangalore-560 066
basavaraj.m@ps.net;
basavarajmj@hotmail.com

Dr. K.C Shet
Professor, Computer Department, National Institute of Technology Karnataka, Surathkal
kcshet@nitk.ac.in; kcshet@yahoo.co.uk

ABSTRACT

Software Cost estimation at activity level is very much accurate than macro estimation with respect to phases of software development life cycle, but the same is very difficult to achieve[1]. Activity based estimation focus on key activities should not be left out and if any effort variance occurs it will be possible to track at particular activity level rather than affecting the entire activities[1]. Activity-based Software estimation based on work break down structure has been explained by collecting and analyzing the data for 12 Enhancements from Application service Maintenance project which were already delivered. This paper explains how to arrive accurate estimation at different micro level activities of Software Development Life Cycle(SDLC).

1. Introduction

Work break down structure(WBS) results in breaking of major component or activity into sub-components or smaller activities. This breaking down process will continue until it is not possible to breakdown each lower lever of sub-components either logically or physically. Each sub-component or smallest activity need to be analyzed and mapped to set of Requirements. WBS suits for most of the Application service maintenance projects since they involve in executing small Enhancements, where we cannot apply the full pledged Estimation Methodology either Function Point Analysis or lines of code(LOC). In this case, most of the company's goes for Activity based software estimation using Work Break down Structure(WBS).

2. Work Break Down Structure

WBS focuses on breaking down project into different activities and assigns efforts to each sub activity. Breaking up of activities into to different activities is not uniform across all the applications or different projects and also varies from one organization to another organization depending on their process defined. There is a need to predict various potential parameters to make WBS more accurate, by analyzing the estimated efforts data of similar projects executed at micro level Activities of SDLC.

3. Work Done

We have taken 12 Enhancements data for simulation which, were delivered for analysis purpose as shown in Fig1. Here Author is not explicitly mentioning the Enhancement names and application names in view of maintaining the company/client confidentiality.

Data collected contains Initial Estimation, Revised Estimation, Approved efforts and Effort variance details.

Effort Variance Details->Six months data							
Sr. No	Enhancement (Enh) Name	Initial Estimation	Revised Estimation	Approved efforts	Approved Vs Initial Estimated - Variance in %	Actual Efforts spent	Actual Vs Initial Estimated - Variance in %
1	Enh1	440	N/A	500	13.64	607	37.95
2	Enh2	160	N/A	300	87.50	276.5	72.81
3	Enh3	168	N/A	211	25.60	119.3	-33.15
4	Enh4	194	N/A	194	0.00	153.6	-14.64
5	Enh5	317	N/A	456	43.85	426.8	31.86
6	Enh6	130	N/A	180	38.46	221.7	16.69
7	Enh7	120	N/A	180	50.00	151.7	68.92
8	Enh8	200	N/A	310	55.00	319.7	59.85
9	Enh9	166	N/A	172	3.61	163.4	5.36
10	Enh10	172	N/A	180	27.91	206.8	15.29
11	Enh11	130	N/A	145	11.54	162.3	24.85
12	Enh12	198	N/A	250	26.26	224.9	19.39

Fig 1. Effort variance details

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 105–110.

Again data has been collected for estimated efforts and actual efforts at different micro level Activities of SDLC for each Enhancement as listed in below tables

Enhancements-Activities Wise Efforts Details

Enhancement(Enh)1

Activities	Estimated Efforts	Actual Efforts	Estimated Efforts In %	Actual Efforts in %
Analysis & Query Resolution	90	101	18.00	16.64
Design	90	97	18.00	15.98
Coding	200	207	40.00	34.10
Testing	40	39.8	8.00	6.56
PM	5	0	1.00	0.00
Quality Assurance(QA)	15	0	3.00	0.00
Reviews	30	82.2	6.00	13.54
UAT(User Acceptance Testing)	15	28	3.00	4.61
Documentation	1	0	0.20	0.00
Config Management	2	0	0.40	0.00
Onsite Coordination	7	13.5	1.40	2.22
Test Case Preparation	5	5	1.00	0.82
Defect Fixing	0	2	0.00	0.33
Environment setup	0	7	0.00	1.15
Implementation/build	0	11	0.00	1.81
Release	0	11	0.00	1.81
Status Meetings	0	1.5	0.00	0.25
Knowledge Transfer	0	1	0.00	0.16
	500	607		

Enhancement 2

Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	90	90	30.00	32.55
Design	35	30	11.67	10.85
Coding	125	124	41.67	44.85
Testing	20	7	6.67	2.53
PM	4	0	1.33	0.00
QA	4	0	1.33	0.00
Reviews	12	9.5	4.00	3.44
UAT	10	0	3.33	0.00
Onsite Coordination	0	6	0.00	2.17
Estimate and Statement of Work(SOW)	0	10	0.00	3.62
	300	276.5		

Enhancement 3

Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	42	54	19.91	45.26
Design	24	16	11.37	13.41
Coding	50	8	23.70	6.71
Testing	24	16	11.37	13.41
PM	4	0	1.90	0.00
QA	9	0	4.27	0.00
Reviews	16	11.3	7.58	9.47
Documentation	22	5	10.43	4.19
Onsite Coordination	8	2	3.79	1.68
Test Case Preparation	12	5	5.69	4.19
Implementation/build	0	2	0.00	1.68
	211	119.3		

Enhancement 4

Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	64	22	32.99	14.32
Design	32	52.3	16.49	34.05
Coding	48	38	24.74	24.74
Testing	24	14	12.37	9.11
PM	8	0	4.12	0.00
QA	8	0	4.12	0.00
Reviews	10	9.8	5.15	6.38
UAT	0	0.5	0.00	0.33
Documentation	0	5	0.00	3.26
Onsite Coordination	0	2	0.00	1.30
Defect Fixing	0	32	0.00	20.83
	194	153.6		

Enhancement 5				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	135	117	29.61	27.41
Design	85	103	18.64	24.13
Coding	65	80	14.25	18.74
Testing	53	77.7	11.62	18.21
PM	20	0.5	4.39	0.12
QA	20	0	4.39	0.00
Reviews	23	18.5	5.04	4.33
Documentation	25	3	5.48	0.70
Onsite Coordination	30	8.3	6.58	1.94
Estimate and SOW(Statement of Work)	0	2	0.00	0.47
Delivery	0	9	0.00	2.11
Follow up	0	4.8	0.00	1.12
Configuration control	0	3	0.00	0.70
	456	426.8		

Enhancement 6				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	40	53	22.22	23.91
Design	40	45	22.22	20.30
Coding	60	65	33.33	29.32
Testing	20	16.5	11.11	7.44
PM	5	6	2.78	2.71
QA	5	0	2.78	0.00
Reviews	10	26.2	5.56	11.82
Documentation	0	6	0.00	2.71
Onsite Coordination	0	1	0.00	0.45
Implementation/build	0	3	0.00	1.35
	180	221.7		

Enhancement 7				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	65	50	36.11	32.96
Design	40	31	22.22	20.44
Coding	35	54	19.44	35.60
Testing	16	5	8.89	3.30
PM	8	0	4.44	0.00
QA	4	0	2.22	0.00
Reviews	4	9.7	2.22	6.39
UAT	8	0	4.44	0.00
Onsite Coordination	0	2	0.00	1.32
	180	151.7		

Enhancement 8				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	70	80	22.58	25.02
Design	50	20	16.13	6.26
Coding	88	59	28.39	18.45
Testing	30	58.1	9.68	18.17
PM	10	1	3.23	0.31
QA	10	0	3.23	0.00
Reviews	10	15.8	3.23	4.94
UAT	0	0.8	0.00	0.25
Documentation	18	12	5.81	3.75
Config Management	0	0.5	0.00	0.16
Onsite Coordination	16	9.5	5.16	2.97
Test Case Prepapartion	8	0	2.58	0.00
Implementation/build	0	29	0.00	9.07
Release	0	18	0.00	5.63
Estimate and SOW	0	8	0.00	2.50
Delivery	0	8	0.00	2.50
	310	319.7		

Enhancement 9				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	50	50	29.07	30.60

Design	24	28.2	13.95	17.26
Coding	24	23.5	13.95	14.38
Testing	32	38.6	18.60	23.62
PM	10	0	5.81	0.00
QA	10	0	5.81	0.00
Reviews	8	9	4.65	5.51
Documentation	8	4.1	4.65	2.51
Config Management	0	4	0.00	2.45
Onsite Coordination	6	0.5	3.49	0.31
Test Case Preparation	0	4	0.00	2.45
Status Meetings	0	1	0.00	0.61
Delivery	0	0.5	0.00	0.31
Follow up	0	17.5	0.00	10.71
	172	163.4		

Enhancement 10				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	40	41	22.22	19.83
Design	50	63	27.78	30.46
Coding	32	35	17.78	16.92
Testing	28	36.5	15.56	17.65
PM	12	5	6.67	2.42
QA	18	0	10.00	0.00
Reviews	12	7.8	6.67	3.77
Documentation	10	8	5.56	3.87
Config Management	0	0.5	0.00	0.24
Onsite Coordination	18	3.5	10.00	1.69
Delivery	0	5	0.00	2.42
Follow up	0	1.5	0.00	0.73
	180	206.8		

Enhancement 11				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	33	48.5	22.76	29.88
Design	26	14	17.93	8.63
Coding	30	40	20.69	24.65
Testing	20	13	13.79	8.01
PM	14	0	9.66	0.00
QA	14	0	9.66	0.00
Reviews	8	6.3	5.52	3.88
UAT	0	9.5	0.00	5.85
Documentation	0	6	0.00	3.70
Config Management	0	2	0.00	1.23
Onsite Coordination	0	21	0.00	12.94
Environment setup	0	2	0.00	8.63
	145	162.3		

Enhancement 12				
Activities	Estimated Efforts	Actual Efforts	Estimated Efforts in %	Actual Efforts in %
Analysis & Query Resolution	90	78.5	36.00	34.90
Design	25	17	10.00	7.56
Coding	25	90.1	10.00	40.06
Testing	38	10	15.20	4.45
PM	11	0	4.40	0.00
QA	11	0	4.40	0.00
Reviews	20	12.3	8.89	5.47
UAT	25	0	11.12	0.00
Documentation	0	13	0.00	5.78
Config Management	0	2	0.00	0.89
Onsite Coordination	5	2	2.22	0.89
Delivery	0	21	0.00	9.34
Follow up	0	0.5	0.00	0.22
	250	224.9		

Fig 2. Enhancements-Activities Wise Efforts Details

4. Results

Out of these 12 listed Enhancements, In estimated efforts, Analysis & Query resolution has come as major for 6 Enhancements, Coding has come as major for 5 Enhancements & Design has come has major for 1 Enhancement. In Actual efforts spent, Analysis & Query resolution has come as major for 5 Enhancements, Coding has come as major for 5 Enhancements & Design has come has major for 2 Enhancements.

Just for illustration purpose line charts have been displayed for three Enhancements, which show estimated Vs Actual effort variance with respect to micro levels of Activities defined in the above tables.

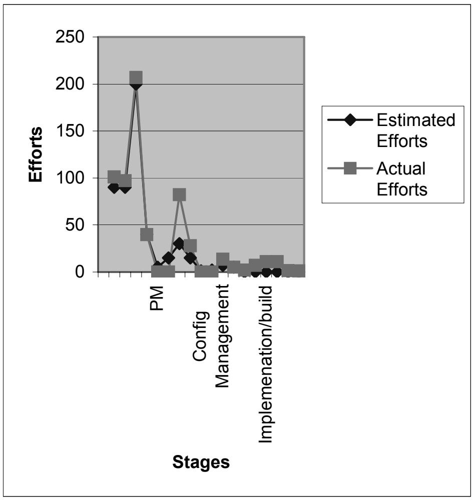


Fig 3. Analysis for Enh1[Line Chart]

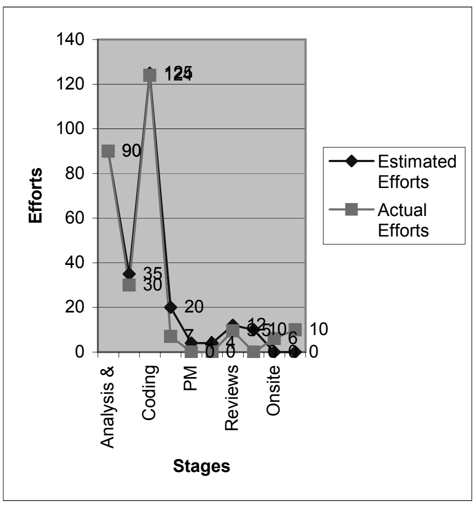


Fig 4. Analysis for Enh2[Line Chart]

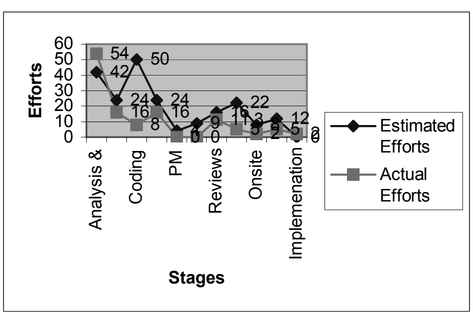


Fig 5. Analysis for Enh3[Line Chart]

By doing similar analysis for remaining Enhancements, it has come to know that, more problematic area is Analysis & Query resolution phase.

Concept of fish bone diagram can be used by identifying root causes 'X's which need to be taken care for estimating next cycle of Enhancements as lessons learnt or by past experience encountered in the respective stage.

This we can put mathematically as

$$Y = F(x1,x2,x3\ldots\ldots\ldots\ldots\ldots xn)$$

Each x1,x2,x3 …. xn is root cause.

In this case Analysis & Query Resolution is having below root cause codes as authors experienced while executing the above listed Enhancements.

1. Analyst capability
2. Query Resolution delay
3. Understanding of requirements
4. Creep in requirements
5. Inadequate inputs from the client

This Activity-based approach also makes an attempt to identify the micro level Activities of SDLC in estimation phase itself to maximum extent to avoid of encountering in actual work.

5. Conclusion

Capturing the metrics on monthly/bimonthly basis & doing the appropriate root cause analysis & fixing the LSL, USL and control charts for all the potential 'X's/Critical 'Xs , it is possible to achieve consistent accuracy to maximum extent in WBS estimation technique. However this does not result in hundred percent accuracy in estimation , but tends to nearer to achieve the accuracy in deploying work breakdown structure estimation technique in future.

6. References

[1] Capers Jones, "Estimating Software Costs", Tata McGraw Hill, Edition 2005

[2] Marc I. Kellner, Raymond J Madachy and David M. Raffo," Software Process Simulation Modelling : Why ? What? How? ", Journal of Systems and Software, Vol. 46, No. 2/3 (15 April 1999)

[3] Kathlen Peters, "Software Project Estimation"

[4] Capers Jones, Assuring Productivity and Quality Applied Software Measurements

[5] Emilia Mendes, Nile Mosely, Steve Counsell, Early Web Size Measures and Effort Prediction for Web Costimation_take Early Web Size Measures and Effort Prediction for Web Costimation_take", Proceedings of the Ninth International Software Metrics Symposium (METRICS'03), 2003 IEEE

[6] Luciana barsei, Sandrao Morasca, and Paolo Paolini, " Estimating the Design Effort of Web Applications", Proceedings of the Ninth International Software Metrics, Symposium(METRICS'03),2003 IEEE

[7] Silvia Abrahao, Geo Poels and Oscar pastor, " Evaluating a Functional Size Measurement Method for Web Applications: An Empirical Analysis", Proceedings of the 10th International ", Proceedings of the 10th International Symposium on Software Metrics (METRICS'04),IEEE

[8] http://sern.ucalgary.ca/courses/seng/621/W98/johnsonk/cost.htm

[9] http://www.geocities.com/shiv_koirala/fp/fp.htm

[10] Function Points Counting Practices Manual Release 4.1.1, IFPUG

[11] J. Brian Dreger ,"Function Point Analysis", Prentice Hall

[12] Roger S. Pressman, "Function Point Analysis – A Practitioner's Approach" , McGraw Hill

[13] http://ourworld.compuserve.com/homepages/softcomp/fpfaq.htm

[14] Capers Jones, Assuring Productivity and Quality Applied Software Measurements

[15] http://www.softwaremetrics.com/fpafund.htm

Software Estimation using Function Point Analysis: Difficulties and Research Challenges

Basavaraj M.J
Perot Systems, Bangalore, Karnataka-India
basavaraj.m@ps.net;
basavarajmj@hotmail.com

Dr. K.C Shet
Professor, Computer Department, National Institute of Technology
Karnataka, Surathkal,
kcshet@nitk.ac.in ; kcshet@yahoo.co.uk

ABSTRACT

Function Point Analysis method serves better efficient way of predicting estimation in beginning phase of software development life cycle(SDLC). Size and complexity of the software can be derived by function point analysis method. Difficulties of estimation using LOC(Lines of Code) can be avoided using Function Point Analysis, since it deals directly with functions or requirements and independent of language or technology. This paper explains how to calculate Function point analysis for the case study Defect Tracking System(DTS) by using function point analysis. Defect tracking system(DTS) case study has been taken from "XYZ" company. In the intention of maintaining confidentiality, authors are not disclosing the company name. Authors also discusses difficulties and challenges by using Function Point Analysis as part of their Research Work.

1. Introduction

Function Point Analysis was developed by Allan J. Albrecht in the mod 1970s. Function points are independent of language, technology, tools and database. Reestimate can be easily achieved by function point since function points are directly linked with change of requirements. Function point analysis can be done at the beginning stage of software development life cycle(SDLC), so that development estimation effort can be predicted at each stage of requirement phase or design phase.

2. Case Study – Defect Tracking System

The Defect Tracking System will be a web application that serves as an aid to managing the testing cycles in software projects. It presents a simple yet effective interface to log defects and provides for workflow management of the defect life cycle right through the typical processes of assigning, fixing and closing a defect.

The DTS will be low cost, and use simple and easily accessible technology, without demanding a high learning curve on the part of the users. The solution shall be low cost, and implemented in quick time. The preferred platform is Microsoft ASP and MS Access. Reporting can be done through MS Excel pivot tables, and thus all the Defect Tracking System should require as installation pre-requisites is a Windows 2000 machine with MS Office installed. A normal desktop running the DTS should serve multiple projects simultaneously.

2.1 Functional Requirements

2.1.1 Testing Defects

2.1.1.1 Enter New Defect : This is the main defect entry screen, the fields that need to be captured are shown in the figure below.

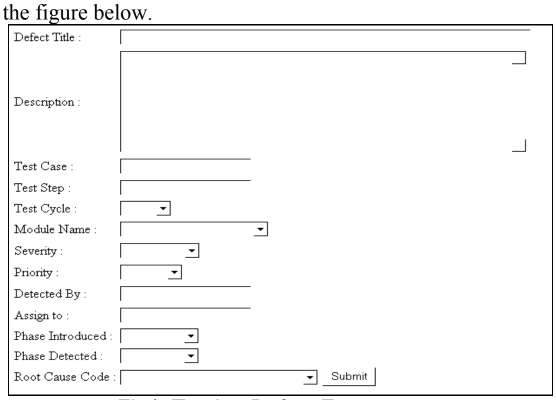


Fig1. Testing Defect Entry screen

2.1.1.2 View Defect ID : In this screen the user enters the Defect ID known to him. On submitting this, if this is a valid defect, the details of the defect are displayed.

2.1.1.3. Query : This screen presents a range of query options. The user can filter the entire list of records for various criteria.

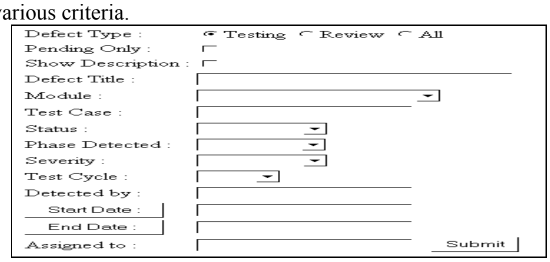


Fig 2. Query screen

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 111–116.

2.1.2 Review Records :

Review Record entry : This Captures the Review Record report.

2.1.2.1 Enter Review Record Header : The fields that should be captured are given below

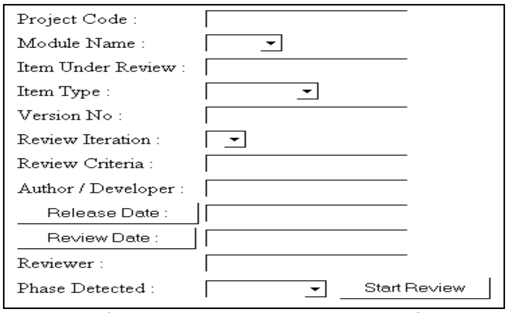


Fig 3. Enter Review record Header Screen

2.1.2.2 Enter Review Defects: The user adds review defects one by one. The fields to be captured are:

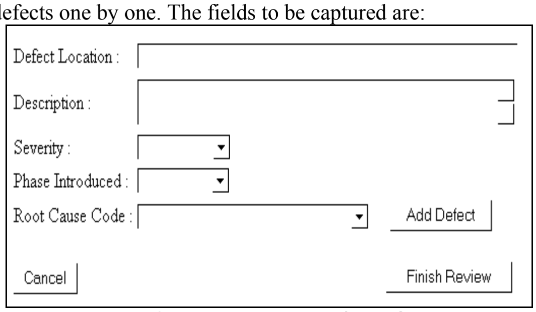


Fig 4. Enter Review Defects Screen

2.1.2.3 Review Record Report : As each record is submitted, the screen should be refreshed so that the review record report is built as shown below. The defects will be automatically assigned to the author. The phase differences and totals of the defect severity are totaled and summarized automatically.

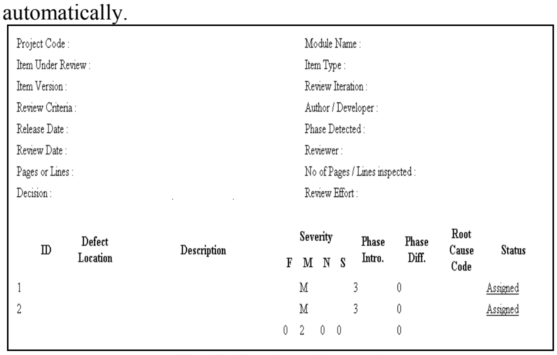


Fig 5. Review Record

2.1.2.4 Review Status : When the user ends the review session, he has to enter the following details:

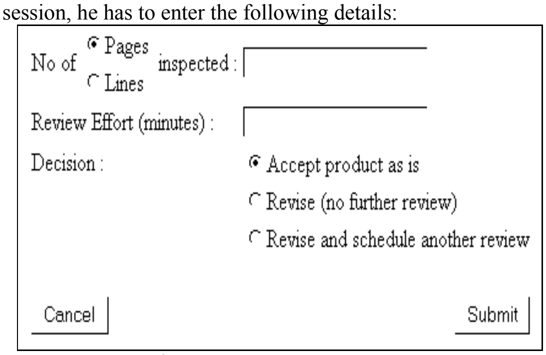


Fig 6. Review status entry screen

2.1.2.5. Review Record View : User enters the ID of a valid review record and it will be displayed

2.1.2.6 Review Record Query : User is given query criteria to select similar to defect query. The query fields are given below.

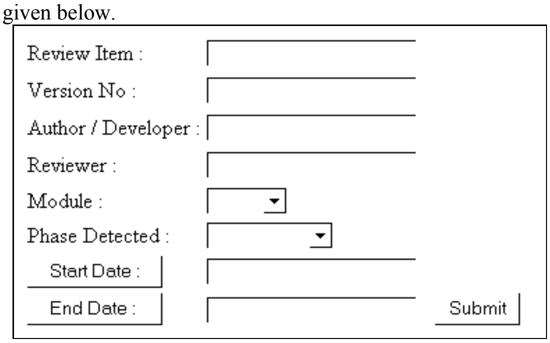


Fig 7. Review Record Query Screen

1. On a successful log in the user will see the test defects entry screen by default.
2. The user should have a navigation menu so that he can access the different screens easily.
3. The user should be able to submit an attachment for a defect, e.g. screen shots, or a detailed write-up.
4. To reduce data entry load of the user, commonly used default values should be populated in the entry fields wherever possible, e.g. today's date.
5. Date entry should be made easy with a calendar popup function.
6. Defects will be of status "New" as soon as entered. The different status changes depending on the defect status like assigned, on hold, dropped, etc.,
7. Defect history should be maintained and tracked.
8. Flexible reports generation using MS Excel
9. There will be an administrator role for DTS. This user can edit defect details after they are committed to the system. He can also edit review records.

2.1.2.7 Defect Status History : Only the administrator can apply status changes. Status changes can be requested by non-administrators. Normally a status change request will be approved by the admin. Also the admin can directly apply a status change. The view of a typical defect status history is given below.

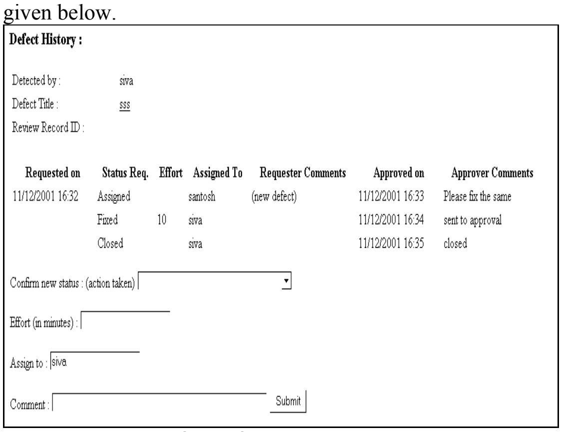


Fig 8 : Defect History screen

3. Analysis of case study

Each screen of the case study – DTS application has been analyzed for counting function Points. Boundary has been identified for Defect tracking system, navigating from start point to end point by mapping to requirements. Physical & logical interfaces are addressed while counting the Function Point.

3.1 FP Counting Procedure

FP counting procedure includes identification of counting boundary, in this case complete DTS. Counting boundary includes Data Function Types count, Transactional Function Types Count which leads to Unadjusted Function FP Count[UAF]. Counting Boundary also includes Value Adjustment factor[VAF] which is determined by General Systems characteristics.

We have arrived Final FP count for case study using below formulae[14].

Final FP = UFP * VAF

Again Data Function type count includes identification of Internal logical files(ILF) and External Interface files(ILF). Transactional Function Type count includes identification of External Inputs, External Outputs, and External Inquiries.

In this case study, we have divided complete application boundary into four modules. Each module we have arrived External Inputs, External Outputs, External Inquiry, Internal logical files(ILF) and External Interface files(ILF), File Type Referenced(FTR) , Record Element Type(RET), and Data Element Type(DET).

3.2 General systems Characteristics (GSC).

We need to consider other things also while making a software, depending on architecture followed, response time Portability, scalability & etc., This turns to external factors which affects software cost. From GSC we can arrive VAF (Value Added Factor). There are 14 points considered to come out with VAF (Value Added factor)[12].

1. Data communications
2. Distributed data processing:
3. Performance
4. Heavily used configuration
5. Transaction rate
6. On-Line data entry
7. End-user efficiency
8. On-Line update
9. Complex processing
10. Reusability
11. Installation ease
12. Operational ease
13. Multiple sites
14. Facilitate change

All the GSC has ratings from 0 to 5.[12]

In our case study implementation, we have assigned a degree of influence (DI) on a 0-5 scale to each GSD. We have arrived Total Degree of Influence (TDI) by summing all DIs for all 14 GSCs.

Degrees of Influence are defined as below[12].

0 Not present, or no influence
1 Incidental or insignificant influence
2 Moderate influence
3 Average influence
4 Significant influence
5 Strong influence throughout

Variable adjustment factor[VAF] is calculated by below formulae[14]

VAF = (TDI * 0.01) + 0.65

4. Results

In case study implementation, we have divided complete application into various sub modules. Each module we have identified External Inputs, External Outputs, External Inquiry, Internal logical files(ILF) and External Interface files(ILF), File Type Referenced(FTR) , Record Element Type(RET), Data Element Type(DET) for each screens by taking into consideration of physical layout of the screen & logical perspective also.

4.1 ILF/ELF Complexity

A functional complexity based on number of DET and RET is assigned to each identified ILF and EIF as below mentioned complexity matrix.

Table 1. Complexity Matrix[DETs/RETs][12]

RETs	DETs		
	1-19	20-50	>=51
1	Low	Low	Average
2-5	Low	Average	High
>=6	Average	High	High

Table 2. ILF/EIF Complexity Matrix[14]

Rating	Values	
	ILF	EIF
Low	7	5
Average	10	7
High	15	10

4.1.1 EI Complexity

A functional complexity based on number of File Type Referenced(FTRs) and Data element types(DETs) is assigned to each identified EI as below mentioned complexity matrix

Table 3. Complexity Matrix[DETs/FTRs][12]

FTRs	DETs		
	1-4	5-15	>=16
1	Low	Low	Average
2	Low	Average	High
>=3	Average	High	High

Table 4. EI Complexity[14]

Rating	Unadjusted FP
Low	3
Average	4
High	6

4.1.2 EO Complexity

A functional complexity based on number of File Type Referenced(FTRs) and Data element types(DETs) is assigned to each identified EO as below mentioned complexity matrix.

Table 5. Complexity Matrix[DETs/FTRs][12]

FTRs	DETs		
	1-5	6-19	>=20
0-1	Low	Low	Average
2-3	Low	Average	High
>=4	Average	High	High

Table 6. EO Complexity[14]

Rating	Unadjusted FP
Low	4
Average	5
High	7

- **EQ Complexity**

A functional complexity based on number of File Type Referenced(FTRs) and Data element types(DETs) is assigned to each identified EQ as below mentioned complexity matrix.

Table 7. Complexity Matrix[DETs/FTRs] [12]

FTRs	DETs		
	1-5	6-19	>=20
0-1	Low	Low	Average
2-3	Low	Average	High
>=4	Average	High	High

Table 8. EQ Complexity[14]

Rating	Unadjusted FP
Low	3
Average	4
High	6

5. FP Calculation Details

Assumptions are arrived from the requirements of the case study stated as below.

In Testing defects Module phase introduced and phase detected taken as hard coded values as

1. Requirements and specification – 1
2. Design – 2
3. Coding and unit testing – 3
4. System Testing – 4
5. Integration testing - 5

Test Cycle value is hard code like 1, 2, and 3

Module name, Severity, Priority and Root cause code values are maintained as static table values.

In View Defect ID module, Defect ID is internally Generating while adding the Defects in Enter New Defect screen.

In Query Module name, status name values are maintained as static table values and Start Date will be made available as Current system Date to the user. Choice is given to the user if he wants to change also.

In Enter review record header module, Project Code and Item Type values are maintained as static table values and Review Iteration is hard coded.

No maintenance screens are to be provided for static tables. Assuming that data need to be entered from Data Base Administrator.

There is no screen shot provided for login screen in the requirement. Assuming that one screen is to be provided as login screen which included user name, password and project. Project codes are maintained as static table.

Testing Defects Module

Enter New defect screen - DET is counted as 15. [Defect Title, Description, Test Case, Test Step, Test Cycle, Module Name, Severity, Priority, Detected By, Assigned to, Phase Introduced, Phase Detected, Root Cause Code, Submit and also Defect ID is generating while entering the details. Here we need to count DET for this also as one].

RET is counted as one since this screen/module is considered as user recognizable subgroup of data elements within an ILF or EIF.

FTR is counted as one since ILF is maintained and read by the EI we need to count as only one.

ILF is counted as one, since this module is maintained within application boundary.

EI is counted as four[Module Name, Severity, Priority, Root Cause Code-All are Maintained in different static tables].

EO is not counted for this screen, since it is not applicable.

EQ is counted as six[Test Cycle, Module Name, Severity, Priority, Phase Introduced, Phase Detected-All drop down boxes].

EIF is counted as one since Item under review and Item Type is referenced by Module name.

In similar way we have calculated FP counts of other modules and the same has been depicted below.

DETAIL SHEET

		Number of			External Input			External Output			External Inquiry			External Interface File			Internal Logical File		
Name of Module within the application boundary	Ref. in Specs	File Type Referenced	Record Element Type	Data Element Type	Low	Average	High	Low	Average	High	Low	Average	High	Low	Average	High	Low	Average	High
Testing Defects	2.1.1.1	1	1	15	4						6			1			1		
	2.1.1.2	0	0	0							1								
	2.1.1.3	0	0	14							1								
Review Records	2.1.2.1	1	1	13	3												1		
	2.1.2.2	0	1	6	2												1		
Report	2.1.2.3	0	0	27					5										
	2.1.2.4	0	1	4													1		
	2.1.2.5	0	1	1							1								
	2.1.2.6	0	0	7	1														
Defect History	2.1.2.7	0	1	15	1														
Count		1	6	75	11	0	0	0	5	0	9	0	0	1	0	0	4	0	0

Fig 9: FP Calculation Sheet-Detail

UNADJUSTED FUNCTION POINTS

Type	Complexity	Count	Multiplier	Subtotal
External Inputs	Low	11	* 3 =	33
	Average	0	* 4 =	0
	High	0	* 6 =	0
External Outputs	Low	0	* 4 =	0
	Average	5	* 5 =	25
	High	0	* 7 =	0
External Inquiries	Low	9	* 3 =	27
	Average	0	* 4 =	0
	High	0	* 6 =	0
Internal Logical File	Low	4	* 7 =	28
	Average	0	* 10 =	0
	High	0	* 15 =	0
External Interface File	Low	1	* 5 =	5
	Average	0	* 7 =	0
	High	0	* 10 =	0
UNADJUSTED FUNCTION POINTS			=	118

PROCESSING COMPLEXITY

No.	General System Characteristics	Degree of Influence	References in Spec
1.	Data Communications	3	
2.	Distributed Data Processing	0	
3.	Performance	1	
4.	Heavily used configuration	2	
5.	Transaction rate	0	
6.	On-line data entry	5	
7.	End-user efficiency	2	
8.	On-line update	2	
9.	Complex processing	0	
10.	Reusability	0	
11.	Installation ease	0	
12.	Operational ease	0	
13.	Multiple sites	0	
14.	Facilitate change	2	
	Total degree of influence	17	

TOTAL FUNCTION POINT COUNT

Value Adjustment Factor =	(Total Degree of Influence * 0.01) + 0.65 =	0.82
Total Function Point Count =	Unadjusted Function Points * Value Adjustment Factor =	96.76

Fig 10: FP Calculation Sheet-Summary[14]

7. Difficulties and Research Challenges

Discussion : Even though Function Point Analysis method is a mature technology in software estimation, major problem lies in how to adjust it for specific application environment for getting accurate and precise measurement. Also Function Point Analysis is based on manually driven method and takes lot of time to calculate function pint counts. Again Expertise involved in arriving Function Point count differs from the non-experienced people in arriving the function point count for the same set of requirements for same specific application environment. Weight ages assigned to Complexities discussed were also under discussion, since same may differ depending on application environment, complexity and nature and type of the project. Also other than General System Characteristics some other factors may also influence or applicable while estimating depending on specific application environment[16]. Authors are addressing this exercise as part of their research to come out with pattern for Function Point Analysis Methodology.

8. Conclusions

In this case study, we have attempted to calculate Function points accurately by identifying external inputs, external outputs, internal logic files, external interface files, , file type referenced, record element type, Data element type for each screens. It is believed that, due to this approach there will be significant improvement in accuracy of predicting estimation in earlier phases of SDLC in case of business applications but it may not applicable for scientific or technical or system applications.

8. References :

[1] R Agarwal, Manish Kumar, Yogesh, S. Mallick, R.M Bharadwaj, D Ananthwar, “Estimating software projects”, Software Engineering Notes vol 26 no4, AGM SIGSOFT, Jul 2001

[2] Emilia Mendes, Nile Mosely, Steve Counsell, Early Web Size Measures and Effort Prediction for Web Costimation_take Early Web Size Measures and Effort Prediction for Web Costimation_take”, Proceedings of the Ninth International Software Metrics Symposium (METRICS’03), 2003 IEEE

[3] Jack E. Matson, Bruce E. Barret, and Joseph M. Mellichanp,”Software Development Cost Estimation Using Function Points”, IEE

[4] Luciana barsei, Sandrao Morasca, and Paolo Paolini, ” Estimating the Design Effort of Web Applications”, Proceedings of the Ninth International Software Metrics, Symposium(METRICS’03),2003 IEEE

[5] Silvia Abrahao, Geo Poels and Oscar pastor, ” Evaluating a Functional Size Measurement Method for Web Applications: An Empirical Analysis”, Proceedings of the 10th International Symposium on Software Metrics (METRICS’04),IEEE

[6] http://sern.ucalgary.ca/courses/seng/621/W98/johnsonk/cost.htm

[7] http://www.geocities.com/shiv_koirala/fp/fp.htm

[8] Function Points Counting Practices Manual Release 4.1.1, IFPUG

[9] J. Brian Dreger ,”Function Point Analysis”, Prentice Hall

[10] Roger S. Pressman, “Function Point Analysis – A Practitioner’s Approach” , McGraw Hill

[11] http://ourworld.compuserve.com/homepages/softcomp/fpfaq.htm

[12] Capers Jones, “Assuring Productivity and Quality Applied Software Measurements”, McGraw-Hill, Second edition

[13] http://www.softwaremetrics.com/fpafund.htm

[14] http://www.softwaremetrics.com/freemanual.htm

[15] Capers Jones, “Estimating Software Costs”, Tata McGraw Hill, Edition 2005

[16] Charles R. Symons, “Function Point Analysis : Difficulties and Improvements” IEEE Transactions on Software Engineering, Vol. 14.No.1,Jan 1988

Distributed Intrusion Detection System for Sensor Networks

Biswajit Panja and Sherif Rashad
Department of Computer Science
Morehead State University, Morehead, KY 40351
{b.panja, s.rashad}@morehead-st.edu

Abstract - **An intruder tries to disable the single point in a network, i.e. the central analyzer. If this is disabled, the entire network is without protection. Since the sensor nodes fail often, the use of a centralized analyzer is highly limited. Processing all the information at a single host implies a limit on the size of the network that can be monitored. Because of the limit of the central analyzer, it is difficult to keep up with the flow of information in large network like sensor. We have proposed distributed intrusion detection system for distribute sensor networks.**

I. INTRODUCTION

In today's world, information has gained an utmost importance. The failure to protect information could result in the loss of people, organizational resources and a great deal of time in attempting to recover it. Intrusion Detection Systems (IDS) help computer systems prepare for and deal with external and internal attacks. Information is collected from a variety of system and network sources, and that is then analyzed so that it adheres to the security required for the system. Vulnerability assessment technologies along with Intrusion Detection allow organizations to protect themselves against network security problems. An Intrusion Detection System (IDS) is a computer program that attempts to perform intrusion detection (ID). There are two well known methods to implement the same. They are misuse detection and anomaly detection. At times a combination of these techniques is used depending on the applications. It is preferred that the IDS perform its tasks in real time. IDs are usually classified as host-based or network-based. In Host-based systems, the decisions on information are obtained from a single host usually called as audit trails [3], while network-based systems obtain data by monitoring the traffic of information in the network to which the hosts are connected. Intrusion detection models in a distributed system can be distinguished as follows:

a. Misuse detection model [5]**:** A specific pattern of events within a network is analyzed to detect that the system has been misused. The sequence of events such as audit trails or systems logs on analysis will depict an unusual pattern. This pattern corresponds to exploitation of weak points within the system.

b. Anomaly detection model [5]**:** Changes in the patterns of behavior of the system is analyzed. Metrics obtained from the system's operation aids in building a model for this type of Intrusion. Events or observations that are intrusive in nature are flagged, if they have a significant deviation from the model used.

Motivation: Intrusion detection is one of the most important aspects in distributed sensor networks. Distributed sensor networks communicate among large numbers of sensor nodes; those are connected remotely or locally. Different types of intrusion are possible in sensor networks. Some of them are as follows:

1. Some sensor nodes from the group compromise the information.
2. Nodes from other groups which are not authorized to get information and join the group with false identification and access the messages.

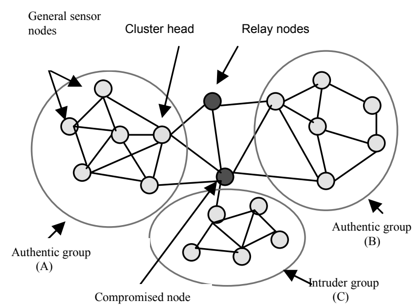


Figure 1. Motivational architecture

In figure 1 group A and B are authentic groups. Group C is nodes. The relay node can compromise and passes the information to unauthentic group. The above figure shows such relay nodes. So the relay node is intruder. Other sensor nodes can also be the intruder, if it passes information to other unauthentic group. The other intrusion possible is if unauthentic nodes of group C can join authentic group A or B by falsifying the identification.

General constraints in sensor networks

- *Hardware:* The hardware constraints for deployment of intrusion detection in sensor networks are memory, power, CPU, radio frequency channel. In modern distributed computers those parameters are not important.
- *Failure:* Sensor nodes fail frequently. The routing changes frequently.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 117–122.

- *Size:* Size of the sensor network devices need to be considered. The nodes and network device need to be small, as sometime it needs to deploy it in secret locations.
- *Scalability:* Sensor nodes are deployed in multiple groups. The number of nodes in each group is large.

Existing intrusion detection systems [2]*:* Many of the existing network and host-based intrusion detection systems perform data collection and analysis centrally using a monolithic architecture (single layered). Audit trails or packets in a network are monitored and collected by a single host, and it is then analyzed by a single module using anomaly-based or rule-based detection techniques. In distributed applications, the patterns from the different distributed modules are collected and the data is sent to a central location where it is analyzed by a monolithic engine.

An intruder tries to disable the single point in a network, i.e. the central analyzer. If this is disabled, the entire network is without protection. Since the sensor nodes fail often, the use of a centralized analyzer is highly limited. Processing all the information at a single host implies a limit on the size of the network that can be monitored. Large numbers of sensor nodes are deployed. After that limit, the central analyzer becomes unable to keep up with the flow of information. Distributed data collection can also cause problems with excessive data traffic in the sensor network. It is difficult to add capabilities to the IDS. Changes and additions are usually done by editing a configuration file, adding an entry to a table or installing a new module. Analysis of network data can be flawed. Performing collection of network data in a host other than the one to which the data is destined can provide the attacker the possibility of performing Insertion and Evasion attacks. We have modified the existing distributed intrusion detection system. We will see if it can fit in distributed sensor network.

Problem Statement: Different types of intrusions are possible in sensor networks. There can be an active intruder and a passive intruder. Active intruders are nodes who come from outside to harm the network, and passive intruders are insider nodes which passes the authentic information to outsiders. Below we have analyzed different possible intrusion in sensor network.

Relay node as intruder: In figure 1 group A and B are authentic groups. Group C is not authentic. The cluster head of group A has the final information of the group. It passes the information to group B through relay nodes. The relay node can compromise and passes the information to the unauthentic group. The above figure shows such relay nodes. So the relay node is the intruder. Other sensor nodes can also be the intruder, if it passes information to other unauthentic group. The other intrusion possible if unauthentic nodes of group C can join authentic group A or B by falsifying the identification.

Cluster head node as intruder: Sensor network works in different clusters, each cluster has a cluster head. In figure 1, each cluster head will have the final data or final decision on behalf of the group. If cluster head passes the information to the unauthentic group C, then it is taken as the intruder.

General sensor nodes as intruder: General sensor nodes can also be intruder because of the following reasons:

1. It can misroute the information or data from its neighboring nodes.
2. Sensor nodes have very limited memory. Some sensor nodes can send unnecessary information to other nodes to make buffer overflow.
3. It can jam the network by not passing the information to other nodes.

Outsider senor nodes as intruder: If some sensor node can falsify the identity and join an authentic sensor group, then it is hard for the authentic group to distinguish between authentic and intruder sensor nodes. After joining a authentic group, that node can pass the information to the unauthentic group.

Constrains of sensor network which make intrusion probability high: Battery Power/Energy, Transmission Range, Memory (Program Storage and Working Memory), Unattended Operations, Ad hoc Networking, Limited Pre-Configuration, Channel error, Unreliable communications, Isolated subgroups, Unknown Recipients.

Intrusion because of less Battery Power/Energy: Sensor nodes having limited battery power. Intruder nodes can falsify the identity and replace the failed authentic node. Implementation of IDS also requires energy, so we need to balance it with the other functions done by the sensor network.

Intrusion because of Transmission Range: Low transmission range of sensor nodes can leads to loosing of information and the intruder nodes can take advantage of that.

Memory: Intrusion is possible by buffer overflow.

Intrusion because of Unattended Sensor node groups: The deployment of sensor network is very dynamic; sensor network can be deployed from aircraft. There can be unattended sensor group near intruder group, and managing module may not be aware of that. If some other sensor nodes send some authentic information this unattended group then the intruder sensor group can get that information.

Intrusion because of Ad hoc Networking: Sensor network is ad hoc in nature. Because of frequently changing position of nodes, intruder nodes get to falsify identification.

Channel error: Wireless sensor networks use radio frequency (RF) channel. This channel has very less capability because of its not physical channel. Intruder nodes can attack the channel and get the information.

Intrusion because of Unknown Recipients: Sensor network contains hundreds and thousands of sensor nodes. It is impossible to keep track of all sensor groups and all sensor nodes. Authentic information can be sent to intruder nodes by mistake.

Intrusion because of Unreliable communications: Communication can fail because of frequently sensor node failure. Intruder nodes can take advantage of this situation and act as authentic node to get the information.

Sub problems: We have divided the problem in four sub-problems. We have divided the sensor network groups in four modules they are, Monitor module, Behavior Collection module, Managing module, and General sensor nodes. We have to analyze the distributed IDS's implications in the sensor network.

1. Implication of Distributed IDS "Monitor module" in Sensor Network:

- Distributed computers have more hardware resources than sensor nodes, so more efficient than sensor network module.
- Data streaming is different in distributed monitor and sensor monitor module, so the detecting properties are different.

2. Implication of Distributed IDS "Behavior collection module" in Sensor Network:

- Sensor network behavior collection modules cannot store much behavior because of memory constraints.
- Isolated sub-groups can have miscommunication with the monitor module because of failure of relay nodes.

3. Implication of Distributed IDS "Managing module" in Sensor Network:

- Frequent routing change of intruder nodes can be hard to handle.
- False alarm could occur because of unknown sensor nodes, though the nodes could be authentic.

4. Implication in General and Intruder sensor nodes:

- False positives and false negatives can occur. False positives are when authorized nodes are identified as intruders, and false negative is when intruder nodes are identified as authentic nodes.

II. RELATED WORK

In [1] the monitoring of intrusions is done by the sensors in a network. There are a number of different systems to detect intrusions. In the signature based intrusion detection systems, the sensors are configured with a number of signatures that are matched against an incoming stream of events. Most systems are initialized to the set of signatures. To update the signature set, the sensor has to be stopped and the execution has to be restarted. Certain types of responses are associated to a known signature. This is a severe limitation when it comes to distributed attacks over a long period of time. The configuration of existing tools is mainly performed manually and is prone to errors. This is especially important if the intrusion detection sensors are deployed in a heterogeneous environment and with different configurations. One major challenge is to determine if the current configuration of sensors is valid for the given environment. In [6] different network based attacks has been explained. The ideas of intrusion detection systems are explained considering operating system and network. "Network-based intrusion detection is challenging because network auditing produces large amounts of data, and different events related to a single intrusion may be visible in different places on the network". State Transition Analysis Technique (STAT) to network intrusion detection has been proposed here. Network based intrusions are modeled using state transition diagrams. In [7] conventional host based and network based intrusion and misuse detection systems have explained. Different IP vulnerabilities have been considered. In [8] Different concepts of intrusion detection are discussed. In [3] Intrusion detection (ID) is defined as "the problem of identifying individuals who are using a computer system without authorization (i.e., 'crackers') and those who have legitimate access to the system but are abusing their privileges (i.e., the 'insider threat')." We add to this definition the identification of *attempts* to use a computer system without authorization or to abuse existing privileges. Thus, our definition matches the one given in, where an intrusion is defined as "any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource." In [4] each mobile node in the network has an intrusion detection system agent running on it all times. This agent is responsible for detecting intrusions based on local audit data. It participates in cooperative algorithms with other intrusion detection system agents. From this it decides if the network is being attacked. Each agent has five modules they are, Local Audit Trial, Local Intrusion Database, Secure Communication Module, Anomaly Detection Modules, and the Misuse Detection Modules. Considering the problems in detecting unknown patterns in Intrusion and its detection and possible approach is using Probabilistic Models [17] that detect intrusions that deviate from known patterns of normal behavior. This is based on *Anomaly Detection.* Initially the system would have to be either trained to data in which intrusions are known or the system is directly run on noisy data, containing intrusions that are detected by its large deviation from other data. One direct assumption in this case is that intrusive data are very different from the normal data. In [16] A variation to the probability distributions using Markov models is anomaly detection over noisy data using learned probability distributions. In this method, the number of normal elements in the data set is significantly larger than the number of anomalous elements. Here a mixture model is used, which separates elements into two cases:

a. an anomalous element with small probability λ and

b. a majority element or normal element with probability (1- λ).

Thus two distributions are possible, a majority distribution (M) and an anomalous

distribution (A). The generative distribution for the data, D, is
$D = (1 - \lambda)M + \lambda A$

For the normal elements we have a function L_M which takes as a parameter the set of normal elements M_t, and outputs a probability distribution, $P_{Mt,}$ over the data D. Similarly we have a function L_A for the anomalous elements which take as a parameter the set of anomalous elements A_t .
Therefore,

$P_{Mt}(X) = L_M(M_t)(X)$ [16]
$P_{At}(X) = L_A(A_t)(X)$ [16]
To detect anomalies, we determine if an element x_t is an anomaly and should remain in M_{t+1} or is an anomaly and should be moved to A_{t+1}. To determine this, we examine the likelihood of the two cases. The likelihood, L, of the distribution D at time t is

$$L_t(D) = \prod_{i=1}^{N} P_D(x_i) = \left((1-\lambda)^{|Mt|} \prod_{x_i \in M_t} P_{Mt}(x_i) \quad \lambda^{|At|} \right) \prod_{x_j \in A_t} P_{At}(x_j) \quad [16]$$

The log likelihood is calculated (LL) at time t as:

$$LL_t(D) = |M_t| \log(1-\lambda) + \sum_{x_i \in M_t} \log(P_{Mt}(x_i)) + |A_t| \log \lambda + \sum_{x_j \in A_t} \log(P_{At}(x_j))$$

The change in LL_t is:

$$M_t = M_{t-1} \setminus \{x_t\}$$
$$A_t = A_{t-1} \cup \{x_t\}$$

If the difference ($LL_t - LL_{t-1}$) is greater than some value c, we state that the element is an anomaly and permanently move the element from the majority set to the anomaly set; else the element remains in the normal distribution. Thus intrusive traces are found out.
If we use the information theoretic measure to measure the regularity of data, *conditional entropy,* we can use the variation in entropy to detect intrusive data. By definition,

$$H(X/Y) = -\sum_{x,y \in X,Y} P(x,y) \log_2 P(x|y) \quad [15]$$

Where P(x,y) is the joint probability of x and y and P(x|y) is the conditional probability of x given y. Using the entropy measure, we can determine the optimal window size for different audit trails, which give the least entropy. That means that this data is more regular with better intrusion detection rate. In [11] Information- theoretic measures can be used for anomaly detection in the following approaches: Measure the regularity of audit data and perform appropriate data transformation. To make the dataset regular, this step is iterated. Determine how to achieve the best performance or the optimal performance/cost trade-off, according to the regularity measure. Use relative entropy to determine whether a model is suitable for a new dataset. This point is particularly important while shifting the approach to a new environment.

III. PROPOSED MODEL

Figure 2 shows our proposed model for intrusion detection. Sensor nodes are deployed in groups. Each group will have part of the responsibility for intrusion detection. Each group acts as a module. There are three modules, monitor module, managing module and behavior collecting module. The responsibility for each module is as follows.
Monitor module: It monitors the system. If it finds unaccepted behavior, then it reports to the managing module.
Behavior collection module: It collects the accepted behavior and keeps the information in its memory. The monitor module compares the monitored behavior with the accepted behavior. The behaviors which do not match are taken as unaccepted behavior.

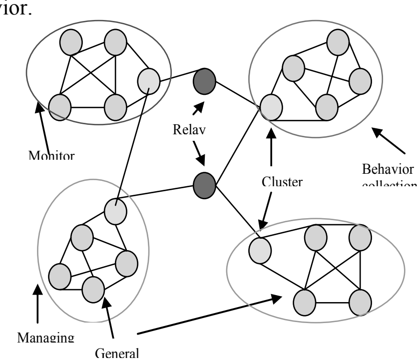


Figure 2. Architecture of proposed model

Managing module: It manages the network from intruder. When monitor module reports occurrence of unaccepted behavior of certain sensor nodes. Then the managing module broadcasts alert message in the network. The message contains the identification of the intruder node.
We have developed an algorithm for each module in IDS of sensor network. We have taken all the constraints into account. The final algorithm is the integrated functions of all the modules. We have shown the block diagram, and then written the algorithm according to the working criteria. The filter filtered the information to get the intrusion related information. Then it sends the data to the monitor module and the monitor module compares the data with the collected data in the behavior collection module.

Algorithm:

```
//Collecting the accepted behavior in behavior collection module
For (Time = initial Time, Time <= final Time, Time++)
{
   If (Network is in working state)
   {
    Collect the accept behavior from the Distributed sensor network
    }
   Else wait till it can communicate
}

//Storing the behavior in different senor nodes in the behavior collection module
 For (Node = node number 1, Node<= final node in the module, Node++)
  If (Memory! = Full)
   {
   Store the rule based behavior in the behavior collection module
   }
  Else switch to different node in that module
```

```
//Collecting information in the filter and deliver the intrusion related information
 While (Network is not in sleep mode)
  {
 Collect the information of the network into the filter
  }
 Filtered the data and deliver the intrusion related information to the monitor module.

//Comparison in monitor module
 While ( Not end of comparison of accepted behavior with behavior collection module)
    {
    Compare the accepted behavior with the behavior collection module
    }
 If (Feels like, it can be an intrusion)
    {
    Decides whether to notify the managing module
    }
 Else keeps working to find intrusion

 Sends the decision to the managing module

//Final decision in managing module
 Managing module takes the final decision
 If (Its an intrusion)
 {
 Broadcasts intrusion alert message
  }
 Else discard the information, and wait for information from monitor module
```

IV. IMPLEMENTATION OF INTRUSION DETECTION TECHNIQUES IN A SENSOR ENVIRONMENT

We use the anomaly-detection technique to detect intrusions in a sensor network to check for behaviors of sensors that deviate from known patterns. We propose two types of detection techniques to achieve this. Both the techniques have their merits and demerits in the sensor network and it would be as per the constraints applicable that we can decide which of these would be most suitable.

The two techniques are using:

a. Probabilistic models
b. Information theoretic measures

***Probabilistic models**:* Here we focus on *Markov Chain Models* to detect intrusions. A discrete-time stochastic process specifies how a random variable changes at discrete points in time. If X_t is a random variable representing the state of a system at time t, where t = 0, 1, 2, … A stationary Markov Chain is a special type of discrete time stochastic process with the following assumptions:

a. the probability distribution of the state at time t+1 depends on the state at time t, and does not depend on the previous states leading to the state at time t;

b. a state transition from time t to time t+1 is independent of time.

Let $P_{i,j}$ be the probability that the system is in a state j at time t+1 given the system is in state t at time t. For a finite number of states, 1, 2, …., s, the stationary Markov chain can be defined by a transition probability matrix:

$$P = \begin{matrix} P_{11} & P_{12} & P_{13} & \dots & P_{1s} \\ P_{21} & P_{22} & P_{23} & \dots & P_{11} \\ . & . & . & . & . \\ . & . & . & . & . \\ P_{s1} & P_{s2} & P_{s3} & \dots & P_{ss} \end{matrix}$$

And the initial probability distribution $Q = [q_1 \;\; q_2 \;\; \dots \;\; q_s]$

Where q_t is the probability that the system is in state i at time 0, and

$$\sum_{j=1}^{j=s} p_{ij} = 1$$

The probability that a sequence of states $X_1, \dots, X_T$ at time 1, …, T occurs in the context of the stationary Markov Chain is computed as follows:

$$P(X_1, \dots, X_T) = q_{x1} \prod_{t=2}^{T} P_{Xt-1Xt} \; [17]$$

The transition probability matrix and the initial probability distribution of a stationary Markov chain is learned from the observations of the system state in the past. Given the observations of the system state $X_0, X_1, X_2, \dots, X_{N-1}$ at time t=0, …, N-1, we learn the transition probability matrix and the initial probability distribution as follows:

$$P_{ij} = N_{ij}/ N_i$$

$$q_i = N_i/ N$$

where

Nij is the number of observation pairs *Xt* and *Xt*+1 with *Xt* in state *i* and *Xt*+1 in state *j*;

Ni. is the number of observation pairs *Xt* and *Xt*+1 with *Xt* in state *i* and *Xt*+1 in any one of the states 1, … , *s*;

Ni is the number of *Xt*'s in state *i*; and

N is the total number of observations.

Over here, the system trains to the data given and builds the stationary Markov chain model of temporal behavior. Let the sequence of audit events be *Et*-(*N*-1)=*t*-*N*+1, … , *Et*, where *E* stands for event. The sequence of states *Xt*-99, … , *Xt* appearing in the window, where *Xi* is the state (the type of audit event) that the audit event *Ei* takes is examined. Using the formula below, the probability that the states occurs in the context of the normal usage is calculated and seen if it supports the sequence of states *Xt*-99, … , *Xt.*

$$P(X_{t-99}, X_{1,\dots,} X_t) = q_{xt-99} \prod_{i=t-98}^{t} P_{Xt-1Xt}$$

The higher probability denotes that the sequence of states results from normal activities. Intrusive events give a low probability of support from the Markov model of the norm profile.

How can the Markov Chain Model be used to the sensor network: The values in the Markov transition matrix are deduced by learning the behavior of the system, i.e. the sensor nodes. The model enlists all transitions that the sensor node has as per its actions. For example, let us consider sensor nodes in a particular group to belong or have the same set of states. If we consider a single sensor node, let us consider that the node is first in a *power off* state. It then makes a transition to the *power on* state due to the action power on. This transition would have a high probability as it is considered normal. The next transition could be a *connect* query to the cluster head. This represents a transition to the *connected* state. There could be innumerous transitions within this *connected* state.

One transition could be the *join query* given by the sensor node to the group. This query would first follow the security protocols before it becomes part of the group. Once the sensor joins the group, the activities of the sensor node are monitored to check for intrusive behavior. Deviations from the studied behavior will give a low probability of support in the Markov transition matrix. Other queries and data could be location tracking data for objects in the sensor's range. All queries pertaining to normal data in this regard would give a higher probability of support. These data could be data related to *object id, object found* and *object tracking* data. From the known patterns of traces on these data, the matrix would give a higher probability of support for data that is more likely to happen.

Considering the type of intrusions possible, we will have to fix a discrete variable on time to analyze other possible intrusion parameters. These could be battery life, memory usage or other implicit parameters of the sensor.

V. Conclusion

We have shown architecture for distributed intrusion detection in wireless distributed sensor networks. The model is distributed and cooperative. The existing intrusion detection algorithm can be used according to the need for intrusion detection. Different possible intrusions have been discussed. Constraints of sensor networks have been explained.

REFERENCES

[1] Designing a Web of Highly-Configurable Intrusion Detection Sensors, Giovanni Vigna, Richard A. Kemmerer and Per Blix Reliable Software Group Department of Computer Science University of California, Santa Barbara, Lecture Notes of Computer Science, Springer 2001.

[2] Computer System Intrusion Detection: A Survey, Anita K. Jones and Robert S. Sielken Department of Computer Science University of Virginia Thornton Hall Charlottesville, VA 2003.

[3] An Architecture for Intrusion Detection using Autonomous Agents, Jai Sundar Balasubramaniyan, Jose Omar Garcia-Fernandez, David Isacoff, Eugene Spafford, Diego Zamboni, COAST Laboratory Purdue University West Lafayette, IN 47907-1398, 2003.

[4] Intrusion Detection Techniques for Mobile Wireless Networks, Yongguang Zhang, Wenke Lee, and Yi-An Huang ACM Mobile Networks and Applications (MONET) Journal 2003

[5] Network Security Essentials: Applications and Standards, William Stallings. Page 292-303

[6] NETSTAT: A Network Based Intrusion Detection System, Vigna, Giovanni. Kemmerer, Richard A. Journal of Computer Security. v 7 n 1 1999. p 37-71.

[7] Identification of Host Audit data to Detect Attacks on Low Level IP Vulnerabilities, Daniels, Thomas E. Spafford, Eugene H. Journal of Computer Security. v 7 n 1 1999. p 3-35.

[8] Intrusion Detection Systems: Expectations, Ideals and Realities, Ranum, Marcus. Computer Security Journal. v 15 n 4 1999. p 25-45.

[9] Securing ad hoc networks, Lidong Z., Zygmunt J. H., IEEE Network, Vol. 13, No. 6, 1999, pp. 24-30.

[10] Mitigating Routing Misbehavior in Mobile, Marti S., Giuli T.J., Lai K. Baker M. Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM 2000, pp 255-265.

[11] Information Theoretic Measures for Anomaly Detection, Wenke Lee, Dong Xiang
Proc. of the 2001 IEEE Symposium on Security and Privacy

[12] Detecting Intrusions Using System Calls: Alternative Data Models, Christina Warrender Stephanie Forrest Barak Pearlmutter Dept. of Computer.IEEE Symposium on Security and Privacy

[13] Learning Patterns from Unix Process, Wenke Lee and Salvatore J. Stolfo Computer Science Department Columbia.Proceedings of the AAAI97 workshop on AI methods in Fraud and risk management, 1997.

[14] Analysis of Distributed Intrusion Detection Systems Using Bayesian Methods, Daniel J. Burroughs, Linda F. Wilson and George V. Cybenko Thayer School of Engineering Dartmouth College Hanover, NH 03755, 2002.

[15] Modeling System Calls for Intrusion Detection with Dynamic Window Sizes, Eleazar Eskin, Wenke Lee, Salvatore J. Stolfo, DISCEX 2001.

[16] Anomaly Detection over Noisy Data using Learned Probability Distributions, Eleazar Eskin Proc. 17th International Conf. on Machine Learning, 2001.

[17] A Markov Chain Model of Temporal Behavior for Anomaly Detection, Nong Ye Arizona State University, 2000

[18] Adaptive Model Generation for Intrusion Detection Systems, Eleazar Eskin, Matthew Miller, Zhi-Da Zhong, George Yi, Wei-Ang Lee, Salvatore Stolfo., 2002.

Numerical Solution to Horizontal Zero-inertia, Viscous Dam-Break Problem

Blaise NSOM, Mathieu JENNY, Youcef ASKOURA and Frédéric PERON

Université de Bretagne Occidentale -. LIME / IUT de Brest. BP 93169. Rue de Kergoat. 29231 BREST Cedex 3 - France blaise.nsom@univ-brest.fr

Abstract : Debris flows such as avalanches and lahars differ from the classical dam-break problem of hydraulics due to the relative importance of viscous versus inertial forces in the momentum balance. An equation of motion describing debris flow in the limit of zero inertia is developed and solved using a converged finite difference numerical, in two limits: short time and long time. These solutions are then combined into a single, universal model.

I. Introduction

Since Ritter's original work on dam-break flow [1], many studies have been performed focusing on experiments, theory and numerical methods (Gill [2]). Dam-break flow has become a classical hydraulic problem with such a large complexity that a higher degree of reproduction of real conditions raises new studies, as certain scenarios of initiation of debris flows, flash floods and lahars can be modelled by dam failures. So, among others, Zanuttigh and Lamberti [3] apply an exact Riemann solution that allows a second-order accuracy of the solution for the power-law section shape to the dam-break problem in valleys with different shapes but the same dam area; Frazao and Zech [4] present an experimental study of a dam-break flow in an initially dry channel with a 90° bend, and successfully compare their measurements of water level and velocity field with numerical results.

Consider a dam obstructing a horizontal smooth channel, dry downstream and with a given quantity of fluid upstream (with height h_0), contained between a fix plate and a dam.

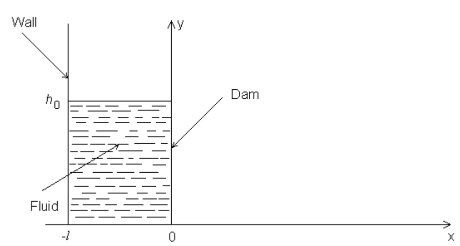


Fig.1: Flow Configuration at negative time

At initial time, the dam collapses and the fluid is released downstream (positive wave), while a negative wave propagates upstream (negative wave). From dam-collapse date to time where negative wave reaches the fix plate, Ritter [1] gives the so-called inertial solution, stating that the wave front advances with a constant speed of $2\sqrt{gh_0}$, while the negative wave moves back with constant speed $\sqrt{gh_0}$.

Between these two extremities, average speed U and the hydrograph are respectively given by

$$U = \frac{2}{3}\left(\frac{x}{t} + \sqrt{gh_0}\right) \qquad (1)$$

$$\sqrt{gh} = \frac{1}{3}\left(2\sqrt{gh_0} - \frac{x}{t}\right) \qquad (2)$$

where the dam is assumed to be located at $x=0$.

This configuration generally represents flow generated by dam failure caused by exceptional rainfall (e.g. Malpasset, France in 1959) or by war action (e.g. Dnieproghes, Ukraine in 1941). The fluid is water and the flow is described by the Navier Stokes and continuity equations, together with the non slip condition. Assuming the shallow water approximation, this system of equations leads to the Barré de Saint-Venant equations (De Saint Venant [5]), a one-dimensional hyperbolic system. The complete hydrodynamic equations describing this unsteady flow in open channel were solved by Faure and Nahas [6], using the method of characteristics. Hunt [7], comparing one-dimensional turbulent flow model down a slope with its viscous counterpart, concluded that the viscous flow model gives the best description for debris flows. Indeed, these flows develop within a long domain, i.e. a domain of space that is much longer than it is wide, so short time behavior described by the previous studies are inappropriate to give a complete description of these natural flows. In the nature, the fluid is generally mud, i.e. a very viscous complex mixture of water with diverse sediments, so the viscous terms are dominant here over the inertial ones. To represent such natural dam-break flow, Nsom et al. [8], Nsom [9] performed an experimental study with glucose-syrup fluids characterized with adjustable viscosity and density. Hunt [7] built similarity solutions for "geological flows" down a sloping 1D channel. Also, Schwarz [10] achieved a numerical study of viscous thin liquid films down an inclined plane. Solving free surface lubrication equations, including the effects of both gravity and surface tension, he states a scaling law for the prediction of finger-width.

In this work, a 1-D model is presented, aiming to provide practical laws, useful to engineers. A priori knowledge of the speed of the flood wave is indeed important because this will determine the available time in which forecast and rescue measures need to be effected. Assuming the shallow-water approximation, equations of motion governing viscous dam-break flow are built and put in non-dimensional form and the initial and boundary conditions are stated. Then, an analytical solution is presented both for short time and long time behavior. Zoppou and Roberts [11] tested the performance of 20 explicit schemes used to solve the shallow water wave equations for simulating the dam-break problem. Comparing results from these schemes with analytical solutions to the dam-break problem with finite-water depth and dry bed downstream of the dam, they found that most of the numerical schemes produce

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 123–128.

reasonable results for subcritical flows. So an explicit procedure was used here, which does not take into account turbulence generated by dam-break wave, as the flow develops over a dry smooth bed (Shigematsu et al. [12]). Numerical results are shown and compared with the analytical ones in each regime.

II. Problem statement

Equations of motion

Let h_0 denote the height of fluid at negative time in a smooth horizontal rectangular channel, g the gravity, ρ and μ the fluid density and viscosity, respectively. Using a cartesian system of coordinates with origin at dam site, x-axis lying on channel-length and z-axis in upwards vertical direction (fig. 1). The fluid is assumed to flow mainly in the direction of x-axis with height h at given control section of abscisse x, at time t. So, the vertical velocities are negligibly small, and therefore the pressure is hydrostatic, the pressure in the flow is given by

$$p=p_0+\rho g(h-z) \tag{3}$$

where p_0 denotes the (constant) pressure at the free surface and g the gravity. The balance between the pressure gradient and the viscous forces is thus expressed by

$$\frac{1}{\rho}\frac{\partial p}{\partial x}=g\frac{\partial h}{\partial x}=\nu\frac{\partial^2 u}{\partial z^2} \tag{4}$$

where horizontal derivatives have been neglected in comparison with vertical derivatives on the right-hand side of (4) because the length of the current is very much greater than its thickness. At the base of fluid layer the no slip condition writes

$$u\left(x,0,t\right)=0 \tag{5}$$

Considering that the shear stress at the top of the current is very much less than its value within the current, and then can be approximated as

$$\frac{\partial u}{\partial z}\left(x,h,t\right)=0 \tag{6}$$

the solution of (4), (5) and (6) is

$$u\left(x,z,t\right)=-\frac{1}{2}\frac{g}{\nu}\frac{\partial h}{\partial x}z\left(2h-z\right) \tag{7}$$

A complete determination of the unknowns u and h requires the equation of continuity which can be written here as

$$\frac{\partial h}{\partial t}+\frac{\partial}{\partial x}\left(\int_0^h u\,dz\right)=0 \tag{8}$$

Substituting (7) into (8) we obtain

$$\frac{\partial h}{\partial t}-\frac{\rho g}{12\mu}\frac{\partial^2(h^4)}{\partial x^2}=0 \tag{9}$$

If l denotes the reservoir length, assume the following set of non dimensional variables:

$$(h',x',x_f',t')=\left(\frac{h}{h_0},\frac{x}{h_0},\frac{x_f}{h_0},\frac{\rho g h^3}{12\mu l^2}t\right) \tag{10}$$

where subscript f denotes wave-front, equation of motion (9) then becomes, in non dimensional form:

$$\frac{\partial^2(h'^4)}{\partial x^2}-\frac{\partial h'}{\partial t'}=0 \tag{11}$$

Eq.11 is similar to the equation of motion obtained by Schwarz [10] and Barthes-Biesel [13], in describing the evolution of a thin liquid layer, flowing down a horizontal plane when surface tension effects can be neglected.

Initial and boundary conditions

Using (10), the fluid height at initial time is given by:

$$h'\left(x',t'=0^-\right)=1 \qquad \text{for} \qquad -1\le x'\le 0 \tag{12}$$

$$h'\left(x',t'=0^-\right)=0 \qquad \text{otherwise} \tag{13}$$

Furthermore, a complementary boundary condition should be imposed upstream, assuming that a short time or an asymptotic solution is sought. These boundary conditions are suggested by experimental observation. For short time case, it is written:

$$h'\left(x'=-l',t'\right)=1 \qquad \text{with} \qquad l'=\frac{l}{h_0} \tag{14}$$

which means that only a given fluid quantity in the upper part of the reservoir is released downstream at the very first instants following dam collapse. While for long time case, it is written:

$$\frac{\partial h'}{\partial x'}\left(x'=-l',t'\right)=0 \tag{15}$$

which means that there is no flow at the fixed wall so, at that site, the free surface is horizontal.

For convenience, in the rest of the paper, non dimensional quantities will be denoted by corresponding capital letters with no primes.

III. Numerical solution

Discretization

To build a numerical procedure, it is necessary to define the channel total length l_t. The non dimensional extreme (downwards) abscissa is

$$L_e=\frac{l_t-l}{h_0} \tag{16}$$

This point is so far from dam site, that the flow is supposed to never reach it during the while of a given experiment (1D assumption), with total duration τ. Then, the problem to solve numerically is described by the following equation of motion

$$\frac{\partial H}{\partial T}=\frac{\partial^2(H^4)}{\partial X^2} \tag{17}$$

associated with the following initial conditions

$$H(X,0)=1 \qquad \text{if} \qquad X\in\left[-L,0\right] \tag{18}$$

$$H(X,0)=0 \qquad \text{otherwise} \tag{19}$$

and boundary conditions

$$\frac{\partial H}{\partial X}(-L,T)=0 \qquad \forall T\ge 0 \tag{20}$$

$$H(Xe,T)=0 \qquad \forall T\ge 0 \tag{21}$$

This problem is solved by a finite difference method.

For this, the function $H(X,T)$ is computed in the set $\left[-L,Le\right]\times\left[0,\tau\right]$, itself discretized in a finite number of identical small rectangles with sides ΔT and ΔX.

The equation will be approximated at grid points located at the following coordinates in the $\left[-L,Le\right]\times\left[0,\tau\right]$ set:

$$\left(X_i,T_j\right)=\left(-L+i\cdot\Delta X,j\cdot\Delta T\right),\ i\in\left[0,\frac{-L+Le}{\Delta X}\right],\ j\in\left[0,\frac{\tau}{\Delta T}\right] \tag{22}$$

Notice that eq.(32) can be put in the form

$$\frac{\partial H}{\partial T}=4H^3\frac{\partial^2 H}{\partial X^2}+3\left(\frac{\partial H^2}{\partial X}\right)^2 \qquad (23)$$

An heuristic approach (Smith [14], Forsythe and Wasow [15], Richtmeyer and Morton [16]) considers the product $\left(4H^3\right)$ in the right-hand side of eq.(23) as a "coefficient of diffusion". Indeed, the following equations are considered :

$$\frac{\partial V}{\partial T}=4\frac{\partial^2 V}{\partial X^2} \qquad (24)$$

and

$$V(X,0)=H(X,0) \qquad (25)$$

The numerical scheme of equations (24-25) is tested using the von Neumann method to provide a stability criterion which is necessary to ensure the convergence of our non-linear problem.

Algorithms

Using Taylor's formula, the derivative of the unknown function can be given by:

$$\frac{\partial H}{\partial T}\left(X,T\right)=\frac{H\left(X,T+\Delta T\right)-H\left(X,T\right)}{\Delta T}-\sum_{n\geq 2}\frac{\Delta T^{n-1}}{n!}\frac{\partial^n H}{\partial T^n}\left(X,T\right) \qquad (26)$$

Also, Taylor's formula can be used to write the non linear term in eq.(17):

$$\frac{\partial^2\left(H^4\right)}{\partial X^2}\left(X,T\right)=\frac{\left[H\left(X+\Delta X,T\right)\right]^4+\left[H\left(X-\Delta X,T\right)\right]^4}{\left(\Delta X\right)^2}\ldots$$

$$-\frac{2\left[H\left(X,T\right)\right]^4}{\left(\Delta X\right)^2}-\sum\frac{\Delta X^{2\left(p-1\right)}}{\left(2p\right)}\frac{\partial^{2p}\left(H^4\right)}{\partial X^{2p}}\left(X,T\right) \qquad (27)$$

Introducing eq.(26) and eq.(27) in eq.(17) produces

$$\frac{H\left(X,T+\Delta T\right)-H\left(X,T\right)}{\Delta T}=\frac{\left[H\left(X+\Delta X,T\right)\right]^4+\left[H\left(X-\Delta X,T\right)\right]^4}{\left(\Delta X\right)^2}\ldots$$

$$-\frac{2\left[H\left(X,T\right)\right]^4}{\left(\Delta X\right)^2}+R_{\Delta X,\Delta T}\left(X,T\right) \qquad (28)$$

with

$$R_{\Delta X,\Delta T}\left(X,T\right)=\sum_{n\geq 2}\frac{\Delta T^n}{n!}\frac{\partial^n H}{\partial T^n}\left(X,T\right)\ldots$$

$$-\Delta T\frac{\Delta X^{2(n-1)}}{\left(2n\right)}\frac{\partial^{2n}\left(H^4\right)}{\partial X^{2n}}\left(X,T\right) \qquad (29)$$

$R_{\Delta X,\Delta T}\left(X,T\right)$ is the residual term which is neglected to solve the numerical problem. Notice that this term can be numerically approximated knowing the solution at the former time step.

Now let

$$H_{i,j}=H\left(X_i,T_j\right) \qquad (30)$$

where X_i and T_j are given by eq.(22), then the finite difference equation to solve, which uses a first order time scheme and a centered second order spatial scheme, is written as

$$H_{i,j}=H_{i,j}+\frac{\Delta T}{\left(\Delta X\right)^2}\left(\left[H_{i+1,j}\right]^4+\left[H_{i-1,j}\right]^4-2\left[H_{i,j}\right]^4\right) \qquad (31)$$

Notice that $H_{0,j+1}$ corresponds to upstream Neumann condition. It is derived from eq.(30), say

$$H_{0,j+1}=H_{1,j+1} \qquad (32)$$

Also, if $i\max$ denotes the maximum value that subscript i can reach, i.e. $i\max$ is rounded off to the integer that is closest to $\frac{L+Xe}{\Delta X}$, then downstream Dirichlet condition yealds

$$H_{i\max,j+1}=0 \qquad (33)$$

$$\frac{\Delta T}{\left(\Delta X^2\right)}\leq\frac{1}{8} \qquad (34)$$

IV. Numerical Results

Free surface profile

The free surface profile is presented in fig.2.

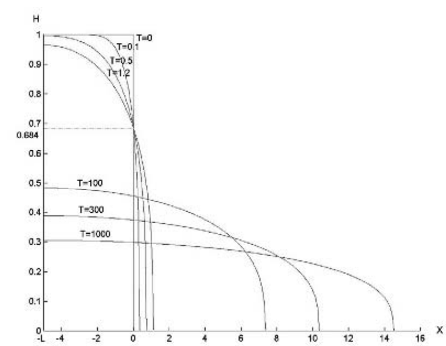


Fig. 2

Fig.2: Time variation of free surface profile for L=10

At large time after dam collapse, it completely differs from Ritter's solution (fig.3), i.e. when the fluid is water, computed using eqs.(1) and (2) which is concave.

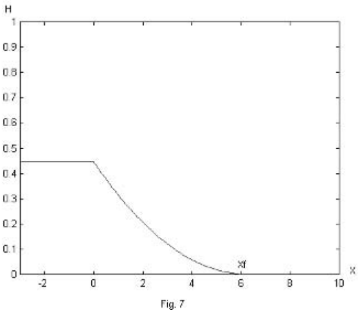


Fig.3: Ritter's solution of inertial dam-break flow

This shows that the convex shape of free surface profile for viscous dam-break flow is intrinsic to the equations of motion governing the problem.

Furthermore, a complete description of the flow should include surface tension, introducing a complementary term in the equation of motion, say

$$\frac{\partial H}{\partial T}=4\frac{\partial}{\partial X}\left[H^3\frac{\partial H}{\partial X}-\frac{1}{B}\frac{\partial^3 H}{\partial X^3}\right] \qquad (35)$$

where B denotes the Bond number, defined as

$$B=\frac{\rho g L^2}{\sigma} \qquad (36)$$

and σ the fluid surface tension. Computation of eq.(54) was carried out using the procedure described in previous section for assigned glucose syrup concentration in water. Fluid physical properties (density, viscosity and surface tension) were taken in Weast et al. [17]. For similar flow configuration, results were quite identical to those obtained from eq.(32), i.e. when surface tension is neglected. In fact, surface tension would affect viscous dam-break flow, only in film lubrication conditions (e.g. Schwarz [10]).

Fluid height

Also, fig.2 shows that during a very short while after dam collapse (short time solution), flow height remains constant at dam site, with

$$H_d(X=0,T)\approx 0.684 \qquad (37)$$

in excellent agreement with analytical solution (eq.20). Viscous solution is characterized by a decreasing of flow height at dam site.

At a given location inside the reservoir, time variation of fluid height is shown in fig.4 which indicates fluid height collapses for stations close to dam site, followed by a smoother decrease for all upstream stations.

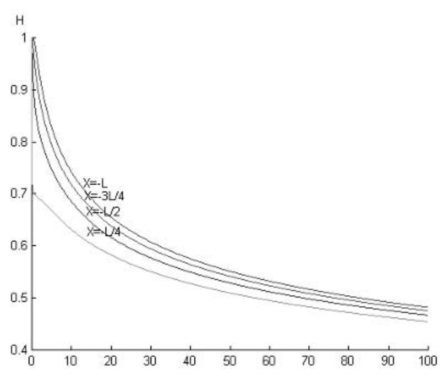


Fig.4: Typical time-height variation at upstream stations

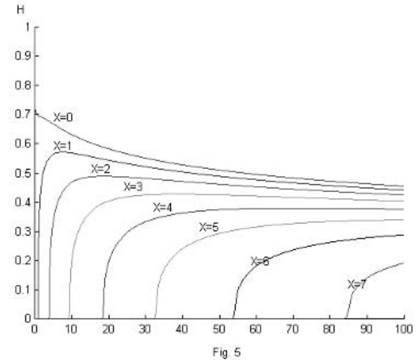


Fig.5: Typical time-height variation at downstream stations

At given downstream station, flow height increases abruptly at first stage, then smoothly to a maximum value and finally decreases after elapsed long time (fig.5).

Front wave position

Time evolution of front wave is presented in fig.6 where three flow regimes can clearly be identified.

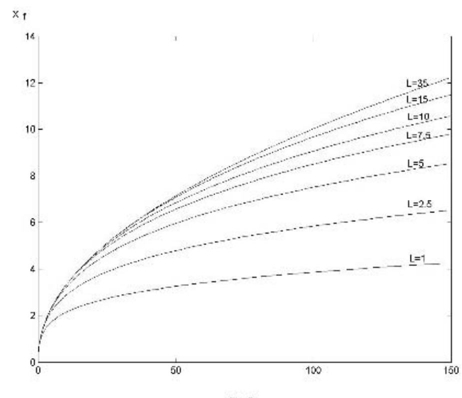


Fig.6: Wave-front evolution for assigned reservoir length

An inertial regime valid immediately after dam collapse, corresponding to $H\leq 0.684$ at dam site and governed by

$$X_f(T)=400T \qquad T\leq 0.004L^{-1} \qquad (38)$$

which meets Ritter's analytical solution.

Then, a viscous regime takes place with the following equation of motion

$$X_f(T)=\sqrt{T} \qquad 0.004L^{-1}<T\leq 0.208L^2 \qquad (39)$$

This regime can be observed during a long time period if the value of the length L grows. This means that the asymptotic solution corresponds to an infinite reservoir length. Indeed, fig. 7 shows that the numerical solution using $L=10$ is in a good agreement with the short time solution until time reach a value around 50.

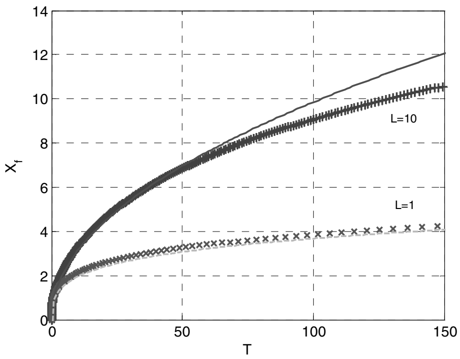


Fig.7 : Comparison between asymptotic and numerical solutions. L=1, dash line (long time asymptotic solution) and cross 'x' (numerical simulation). L=10, full line (short time asymptotic solution) an cross '+' (numerical simulation).

The model predicts that this asymptotic solution is valid for T<20.8. This threshold is then numerically confirmed. The numerical simulation doesn't last enough time to reach the long time asymptotic solution. Nevertheless, when L is set to 1, the long time solution is reached quite early (fig. 7) in order to be observed within the duration of the simulation.

It should be stressed that the reservoir extent L is not involved in short time solution as it holds only near the front moving downstream, while the long time asymptotic solution depends on L, as it holds everywhere.

Nsom [9] obtained experimentally, a Ritter's type evolution of the front $(T<0.2)$; in the viscous regime, a scaling law with the form

$$X_f \propto T^{1/2} \tag{40}$$

and the following equation of motion for the asymptotic regime

$$X_f = 1.86T^{1/5} - 0.902 \tag{41}$$

These results agree well with the numerical results obtained in present work. Also, Huppert [18] obtained a concordant analytical asymptotic solution in the following dimensional form

$$x_f \equiv E.t^{1/5} \tag{42}$$

where E , is a constant depending on the fluid physical properties.

Meanwhile, a comparison with other previous experimental and numerical results is difficult because in the configuration generally considered, the viscosity is neglected (e.g. Martin and Moyce [19], Stanby et al. [20], Shigematsu et al. [12]).

Front wave velocity

The front wave velocity is obtained from the derivation of $X_f(T)$. The numerical result is computed using the following centred second order scheme :

$$U_f\left(T+\Delta T/2\right) = \frac{X_f\left(T+\Delta T\right) - X_f\left(T\right)}{\Delta T} + o\left(\Delta T^2\right) \tag{43}$$

where U_f is the front velocity. The asymptotic velocity is directly calculated from equations (17) and (30). So, we can write :

$$U_f\left(T\right) = \frac{1}{2}\sqrt{\frac{2c}{T}} \tag{44}$$

for short time and

$$U_f\left(T\right) = \frac{1.8625}{5} T^{-4/5} \tag{45}$$

for long time.

The velocity is computed setting $L=10$ and compared to the short time asymptotic solution
(eq. 63). Fig.8 shows that the time evolution of the velocity, obtained using computational resolution, is very similar to the tendency calculated from the asymptotic model for $T<150$.

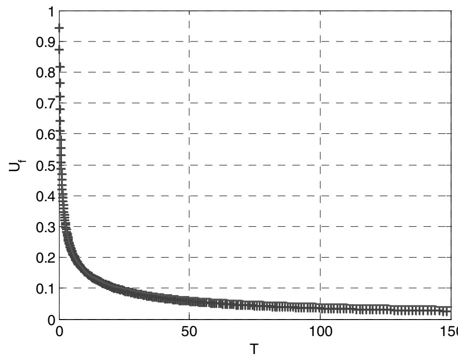


Fig.8 : Front velocity vs time (L=10). '+' : numerical solution. Full line : short time asymptotic solution.

The difference between the value of the velocity given by the numerical and asymptotic models in fig.9 shows that the short time solution differs from the numerical solution up to 5% when $T>20.71$ which is in very good agreement with the expected value (20.8).

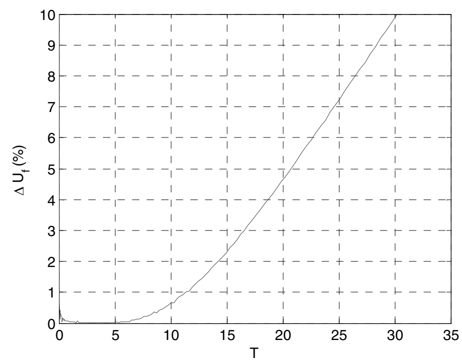


Fig.9 shows also that the short time solution is less sharp for very short time as the error coefficient grows below $T=2$ when the time decreases.

Conclusion

The horizontal viscous dam-break flow was considered. The channel was smooth and dry at initial time. Applying the conservation of mass and momentum with the shallow water approximation, an equation of motion was derived and made non dimensional, when the viscous forces were assumed to be the dominant ones. Then, an analytical solution of the equation of motion was built. Immediately after the dam collapse, an inertial regime takes place. In this regime, flow height at dam site has a (fix) characteristic value and the equation of motion of front wave is of Ritter's type. As flow height inside the reservoir reaches this characteristic value, a short time viscous regime holds with no influence of reservoir extend on front wave equation of motion. Finally, a long time viscous regime takes place depending on reservoir length. Also, a convex free surface profile with a self-similar form and time variation of flow height at both upstream and downstream stations were furnished.

Furthermore, the problem was considered numerically. The previous equation of motion was of porous medium type and was approximated using an explicit finite difference method.

The stability and convergence of the computations were insured using a criteria based on heuristic approach. The very good agreement between numerical and asymptotic solutions shows the consistence of the numerical scheme for both short time and asymptotic form of long time solution which correspond respectively to long and short reservoir length L..

ACKNOWLEDGEMENT

This work was carried out within InterMareC european project. The authors would like to thank Mr Eric Vandenbroucke, associate-director of Technopôle Brest Iroise and Gildas Borel, director of Nasca Geomarine for their help and support.

REFERENCES

[1] A. Ritter „Die fortpflanzung der wesser wellen." Ver Deutsch Ingenieure Zeitschrift, vol. 36(33), pp. 947-954, 1892 (in German)

[2] M. A. Gill, "Dam-break problem." *Encyclopedia of fluid mechanics*, 6, N.P. Cheremisinoff, eds., Gulf, Houston (Texas), 1429-1473, 1987

[3] B. Zanuttigh and A. Lamberti, "Dam-Break Waves in Power-Law Channel Section." J. Hydr. Eng. (ASCE), vol. 127(4), pp. 322-326, 2001

[4] Frazao, S.S. and Zech, "Dam-Break in Channels with 90° Bend." J. Hydr. Eng. (ASCE), vol. 128(11), pp. 956-968, 2002

[5] B. De Saint-Venant, « Théorie du mouvement non permanent des eaux » , C. R. Acad. Sci., vol. 73(3), pp. 147-154, 1871 (in French)

[6] Faure, J. And Nahas, N., "Etude numérique et expérimentale d'intumescences à forte courbure du front. », La Houille Blanche, vol. 16(5), pp. 576-587, 1961 (in French)

[7] B. Hunt, "Newtonian fluid mechanics treatment of debris flows and avalanches", J. Hydr. Div. (ASCE), vol. 120(12), 1350-1363, 1994

[8] B. Nsom, K. Debiane, and J. M. Piau, "Bed slope effect in the dam-break problem", J. Hydr. Res. (IAHR), vol. 38(6), pp. 459-464, 2000

[9] B. Nsom, "Horizontal Viscous Dam-Break Flood: Experiments and Theory." J. Hydr. Eng. (ASCE), vol. 128(5), pp. 543-546, 2002

[10] L. W. Schwarz, „Viscous flows down an inclined plane: Instability and finger formation.", Phys. Fluids A, vol. 1(3), pp. 443-445, 1989

[11] C. Zoppou, and S. Roberts, "Explicit Schemes for Dam-Break Simulations" J. Hydr. Eng.(ASCE), vol. 129(1), pp. 11-34, 2003

[12] T. Shigematsu, P.L.F. Liu, and K. Oda, "Numerical modeling of the initial stages of dam-break waves*."* J. Hydr. Res. (IAHR), vol. 42(2), pp. 183-195, 2004

[13] D. Barthes-Biesel, « Rectification d'un film liquide sous l'effet de la gravité et de la tension superficielle » <http://www.enseignement.polytechnique.fr/profs/informatique/Georges.Gonthier/pi98/film.html>, June 1, 1998 (in French)

[14] G.D. Smith, *Numerical Solution of Partial Differential Equations*, Oxford University Press, New York, 1969.

[15] G.E. Forsythe and W.R. Wasow, *Finite-Difference Methods for Partial Differential Equations*, John Wiley and Sons, New York, 1967

[16] R.D. Richtmyer, and K.W. Morton, *Difference Methods for Initial Value Problems*, John Wiley and Sons, New York, 1967

[17] R.C. Weast, M.J. Astle and W.H. Beyer, eds., *Handbook of chemistry and physics*, CRC, Boca Raton (Fla), 1987

[18] H.E. Huppert, "The propagation of two-dimensional and axisymmetric viscous gravity currents over a rigid horizontal surface", J. Fluid Mech., vol. 121, pp. 43-58, 1982

[19] J.C. Martin, and W.J. Moyce, "An experimental study of the collapse of liquid columns on a rigid horizontal plane", Phil. Trans. Roy. Soc. Lond. A, vol. 244(882), pp. 312-324, 1952

[20] P.K. Stanby, A. Chegni and T.C.D. Barnes. "The initial stages of dam-break flow", J. Fluid Mech., vol. 374, pp. 407-424, 1998.

PIXEL & FEATURE LEVEL MULTIRESOLUTION IMAGE FUSION BASED ON FUZZY LOGIC

Bushra N. Kayani[1], Anwar Majeed Mirza[2], Ajmal Bangash[1], Haroon Iftikhar[3]
Faculty of Computer Science[1], Department of Computer Science[2], Department of Electrical Engineering[3]
GIK Institute of Engineering Sciences and Technology[1], National University[2], University of Ottawa[3]
Topi[1], Islamabad[2], Ottawa[3]
Pakistan[1-2], Canada[3]

Abstract: **The motivation behind fusing multi-resolution images is to create a single image with improved interpretability. In algorithm (based on pixel and feature level) presented in this paper, images are first segmented into regions with fuzzy clustering and are then fed into a fusion system, based on fuzzy "if-then" rules. Fuzzy clustering offers more flexibility over traditional strict clustering; thus allowing more robustness as compared to other segmentation techniques (e.g. K-means clustering algorithm). A recently proposed subjective image fusion quality evaluation measure known as IQI (Image Quality Index) [1] is used to measure the quality of the fused image. Results and conclusion outlined in this paper would help explain how well the proposed algorithm performs**.

Keywords: Image Fusion, Discrete Wavelet Frame Transform (DWFT), Fuzzy C-Mean Clustering, Discrete Wavelet Transform (DWT) and Image Quality Index (IQI).

I. INTRODUCTION

In the past few years multi-sensor systems have utilized data fusion to get the improved and enhanced versions of acquired data. This can be useful in some diagnostic and research situations. Image fusion is a process by which several registered images or some of their features are combined together in such a way that there is no loss of information and introduction of distortion [2]. Fused image produced is thus more suitable and enhanced for human / machine perception. Composite image improves image content and make it easier for detection, recognition and identification of targets and thus increase situational understanding.

Image fusion has a number of applications in remote sensing, multi-focus camera applications, medical imaging, concealed weapon detection and night-time security.

Image fusion is generally performed at three different levels of information representation; these are pixel level, feature level and decision level [2]. Fusing images at pixel level means to perform integration at a level where the pixels are least processed. Each pixel in the fused image is calculated from pixels in the source images by for-example averaging. Fusion at feature level first requires extraction of features from the source images (through e.g. segmentation); fusion then takes place based on features that match some selection criteria. At symbol level/decision level, the output from the initial object detection and classification using source images is then fed into the fusion algorithm. Every image fusion algorithm is performed at one of these three levels or some combination thereof. Our algorithm focuses on a framework which combines aspects of both pixel and feature level image fusion.

Looking in the literature, we find image fusion techniques which vary from simple pixel averaging to complex methods involving principal component analysis (PCA) [4], pyramid based image fusion [3] and wavelet transform (WT) fusion [5]. Lately methods involving wavelet based fusion have gained much popularity because wavelet transform provides directional information while the pyramid representation doesn't introduce any spatial orientation in the decomposition processes. Liu et al [7] proposed a method in which authors take discrete wavelet frame transform (DWFT) of images because of its shift invariant property which lacks in DWT (discrete wavelet transform). Images are then segmented into regions using k-means clustering algorithm and are fed into the fuzzy fusion system.

In this paper an image fusion algorithm based on fuzzy logic is introduced. The segmentation of images is done using fuzzy c-mean clustering algorithm instead of k-means clustering algorithm because fuzzy clustering not only preserves but also emphasizes the grey level present in the image.

The subsequent sections of this paper are organized as follows. Section 2 explains the components of algorithm proposed. Section 3 is about the proposed scheme of image fusion Section 4 explains IQI, an objective image fusion quality evaluation measure followed by results in Section 5 and conclusion.

II. INGREDIENTS OF PROPOSED SCHEME

The key constituents of our scheme are discrete wavelet frame and fuzzy c-mean clustering. Below we have discussed some of the prospects regarding these techniques.

A. Why DWFT?

The lack of translation invariance together with rotation invariance is the key drawback of DWT in feature extraction. Due to shift variance the fusion methods using DWT lead to unstable and flickering results. This can be overcome with DWFT by calculating and retaining wavelet coefficients at every possible translation of convolution filters or in other words the redundant transforms. More detail can be found in MATLAB wavelet toolbox, where it is called Discrete Stationary Transform (SWT).

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 129–132.

B. Fuzzy c-mean Clustering:

Fuzzy c-means is a technique for clustering which allows one data item to belong to more than one cluster, whereas in hard (k-means) clustering an entity belongs to one and only one cluster. For more information on k-means clustering you may visit [11].

In fuzzy c-mean clustering algorithm random membership values and cluster centers are assigned. The iterative process continues to calculate the new membership values and cluster centers according to the distance b/w entities and centers. This process comes to stop when a maximum number of iteration is reached or an objective function reaches a required threshold value [6].

III. Proposed Image Fusion Scheme

It is important to know for the readers that the set of images used in this algorithm are registered images. With registration we find correspondence between images. . It is necessary because only after it is ensured that spatial correspondence (information from different sensors can be guaranteed to come from identical points on inspected object) is established, fusion makes sense. More detail on image registration can be found in [8], [9].

A. Algorithm and Flowchart:

The general framework of the proposed algorithm can be shown with the help of a flowchart. Step # refers to the steps of the algorithm mentioned below.

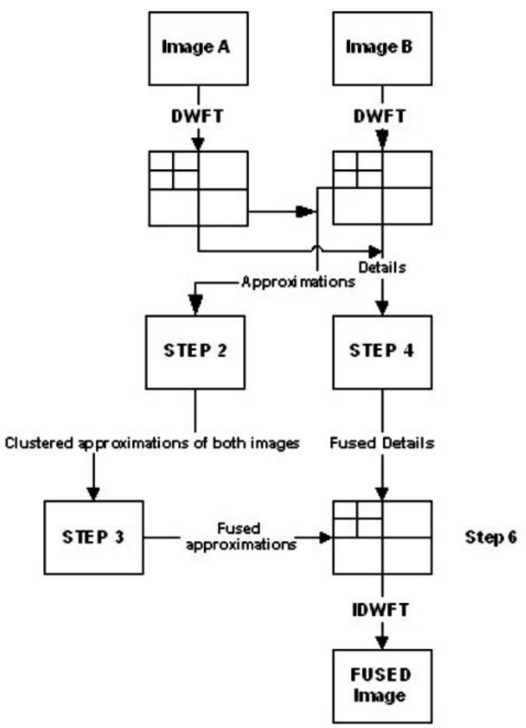


Fig. 1. Flowchart of the proposed scheme

1. Apply DWFT to two registered source images giving detail sub-bands and approximation sub-band.
2. Fuzzy c-mean clustering algorithm is used to segment the approximations into three regions, important region, sub-important region and background region, named on the basis of grey levels. Each pixel will have a degree of membership for the regions it belongs ranging between 0-1.
3. **Feature Level Fusion:-** Segmented approximations are fed into a fusion system based on fuzzy "if-then" rules, to get fused approximations. The membership functions, rules and de-fuzzification function details can be found in [7].
4. **Pixel Level Fusion:-** The details are fused by absolute maximum coefficient selection method.
5. Apply morphological filtering Zheng et al. [4] which use "fill" and "clean" operators to sweep isolated points.
6. With fused Approximations and fused details get fused wavelet frame coefficients map. Take Inverse Discrete Wavelet Frame Transform (IDWFT) and get fused image.

IV. Image Quality Index

Image Quality Index proposed by Piella et al [1] has been used as an objective image fusion quality evaluation measure. IQI is recently proposed, easy to calculate and is used quite often for image quality measurement. The expression of global image quality index is:-

$$Q_o(A,B) = \left(\frac{\delta_{AB}}{\delta_A \delta_B}\right)\left(\frac{2\overline{AB}}{(\overline{A}^2 + \overline{B}^2)}\right)\left(\frac{2\delta_A \delta_B}{\delta_A{}^2 + \delta_B{}^2}\right) \quad (1)$$

Where δ_A is variance of A, δ_{AB} is covariance of A and B and $\overline{A}$ is the mean of A. The value of $Q_o \in [0, 1]$, $Q_o = 1$ means A and B are completely identical.

We then compute λ, a local weight giving more importance to one of the two images. The more the value of λ the more weight is being given to that particular image. To compute the value of λ we have:-

$$\lambda = SF(A)/[SF(A) + SF(B)] \quad (2)$$

In (2) SF is the spatial frequency of image and it measures the overall activity level of the image [10].

$$Q_F = \lambda Q_o(A,F) + (1-\lambda) Q_o(B,F) \quad (3)$$

V. Results And Discussions

Three existing image fusion schemes are used for comparative analysis of our proposed scheme. These schemes are:-

A. DWT based image fusion:-

In this method, images are first decomposed using DWT. Approximation and detail sub-bands are fused by choosing maximum wavelet coefficients from both the DWT coefficients of source images. Fused image is acquired by applying Inverse DWT [5].

B. aDWT based image fusion:-

An advanced wavelet transform (aDWT) method that incorporates principal components analysis and morphological processing into a regular DWT fusion algorithm.[4].

C. Image region fusion using k-means clustering:-

In this method images are first segmented into three regions (Important, sub-important and background) using k-means clustering algorithm. These regions are then fused using fuzzy inference system [7].

In all these schemes images are decomposed to 3rd level and wavelet named 'db1' is being used. This fusion algorithm is done using MATLAB.

A large set of images with 256 grey-levels is tested on proposed scheme and the schemes mentioned above. Both subjective and objective results are shown below in the form of bar chart and fused images.

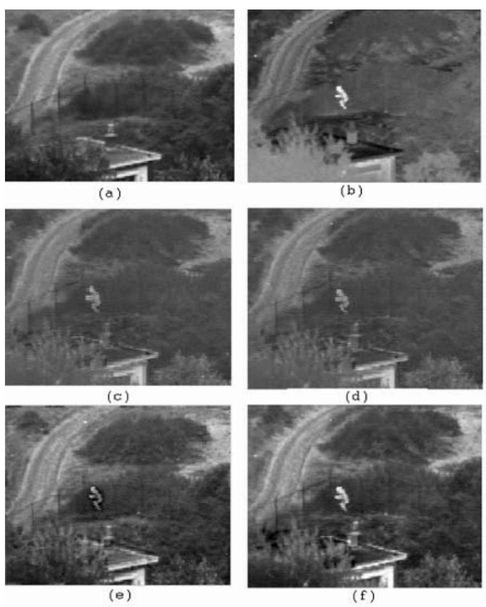


Fig. 2. Fusion of Night time images: (a) Visual Image (b) Infra-red Image (c) DWT fusion (d) aDWT fusion (e) k-means clustering based fusion (f) Proposed scheme

Figures 2, 3, 4 and 6 show subjective fusion quality measures where figure-5 demonstrates quantitative and objective measure for quality measurement of fused image where we have shown the improvement with the help of a bar chart. Proposed algorithm (fuzzy c-means) is being compared with the already existing techniques mentioned in Section 5. These are the average IQI values tested over more than 72 image pairs belonging to different categories.

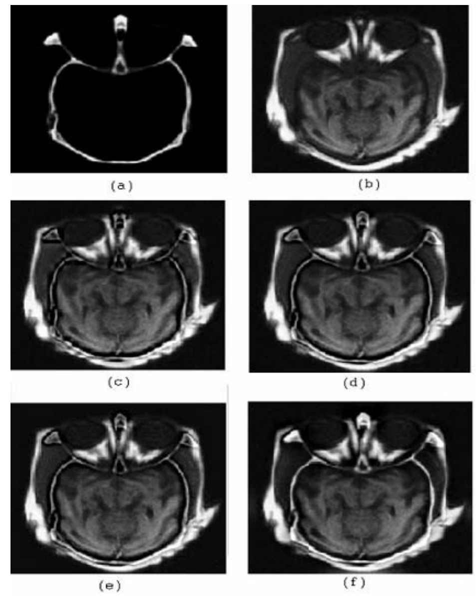


Fig. 3. Fusion of Medical images: (a) CT image (b) MRI Image (c) DWT fusion (d) aDWT fusion (e) k-means clustering based fusion (f) Proposed scheme

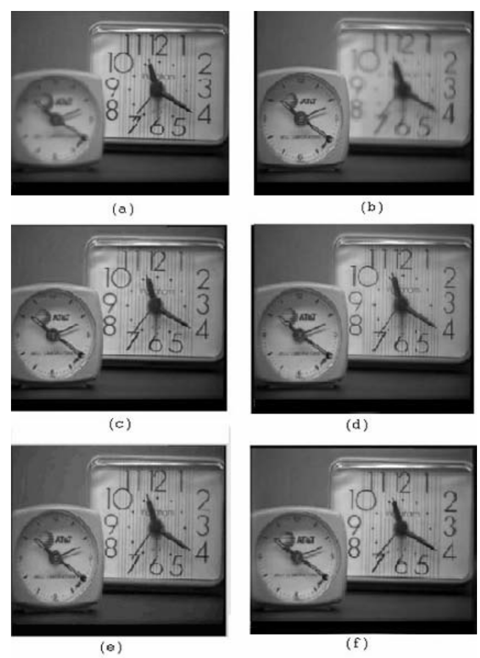


Fig. 4. Fusion of Multi-focus images: (a) Right in focus image (b) left in focus image (c) DWT fusion (d) aDWT fusion (e) k-means clustering based fusion (f) Proposed scheme

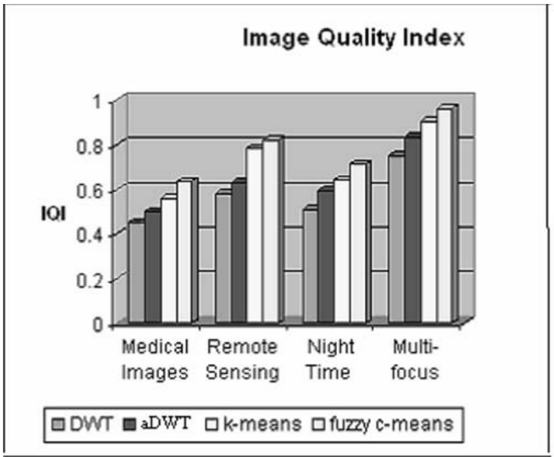


Fig. 5. Average IQI

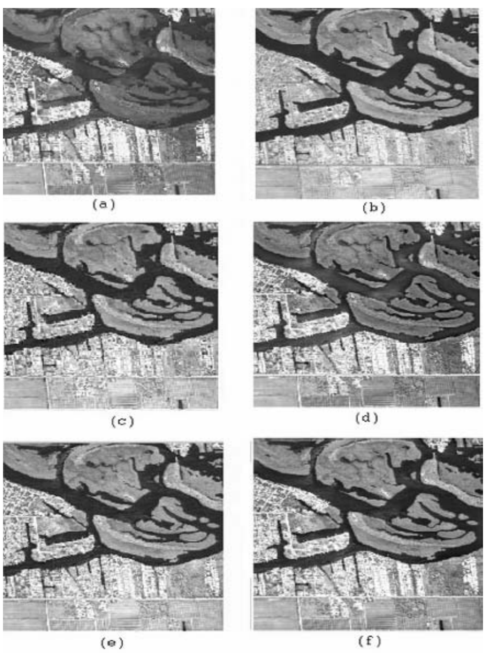


Fig. 6. Fusion of Multi-focus images: (a) Right in focus image (b) left in focus image (c) DWT fusion (d) aDWT fusion (e) k-means clustering based fusion (f) Proposed scheme

VI. CONCLUSION

An image fusion algorithm, based on fuzzy clustering and discrete wavelet frame transform is proposed in this paper. With experimental discussion we conclude that our algorithm improves quality of fused image as compared to fusion algorithm based on k-means clustering algorithm.

Proposed algorithm has one drawback when it comes to computational time. Due to discrete wavelet frame the physical computations required are more and hence the time.

Future work includes the computationally fast image fusion scheme, which efficiently gives good results.

REFERENCES

[1] G. Piella and H. Heijmans, "A new quality metric for Image Fusion" in *International Conference on Image Processing.* 2003, Barcelona, Spain.

[2] Gonzalo Pajares, Jesús Manuel de la Cruz, "A wavelet-based Image Fusion Tutorial" in *Pattern Recognition,* vol 37, no. 9, pp. 1855-1872, 2004.

[3] A.Toet, "Image fusion by a ratio of low pass pyramid" in *Pattern Recognition Letters,* vol.9, no.4, pp. 245-253, 1989.

[4] Yufeng Zheng, Edward A. Essock and Bruce C. Hansen, "An Advanced Image Fusion Algorithm Based on Wavelet Transform – Incorporation with PCA and morphological Processing" in *Proceedings of the SPIE,* vol 5298, pp. 177-187, 2004.

[5] H.Li, S.Manjunath and S.K.Mitra, "Multi-sensor image fusion using the wavelet transform" in *Graphical Models and Image Processing,* vol.57, no.3, pp. 235-245, 1995.

[6] S.R. Jang, C.T. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence,* Prentice Hall Inc, USA, 1997.

[7] Liu Gang, Jing Zhong-liang, Sun Shao-yuan, "Multi resolution image fusion scheme based on fuzzy region feature," in *Journal of Zhejiang University Science A,* vol 7, no. 2, pp. 117-122.

[8] R. K. Sharma and Misha Pavel, "Multi-sensor Image Registration" in *SID Digest Society for Information Display.* Vol xxviii, May 1997, pp.951-954.

[9] Brown, L.G, "A survey of Image registration techniques" in *ACM Computing Survey* vol 24, pp. 325-376, 1992.

[10] Shutao Li, James T. Kwok Yaonan Wang, "Combination of images with diverse focuses using spatial frequency" in *Information fusion,* vol 2, pp. 169-176, 2001.

[11] http://mathworld.wolfram.com/KmeansClusteringAlgorithm.html.

Approximate Solution to the Diffusion-Reaction Problem with Nonlinear Kinetics in Transient Systems

Peralta Reyes E., Regalado Méndez A., Vidriales Escobar G. and González Rugerio C. A.
Universidad del Mar.
Ciudad Universitaria s/n, Puerto Ángel, San Pedro Pochutla, Oaxaca, Méx. CP 70902,e-mail: cagr@angel.umar.mx

ABSTRACT

A method to obtain the approximate solution to the diffusion-reaction problem with nonlinear kinetics in transient systems is presented. The analytical solution to the equation that governs the process is based on the linearization of the kinetics expression through the Taylor series expansion above the surface particle concentration of the key component, which includes a critical radius to avoid negative concentration values. The present results for the average concentration were compared with the numerical solution of the exact problem and the error was less than ten percent for the power-law and Monod kinetics equation.

Key words: Approximate method, Diffusion-reaction problem, Linearization, Critical radius, Dead zone.

I. INTRODUCTION

The evaluation of the global reaction rate is important for the analysis, design and simulation of heterogeneous chemical reactors. However, it is a difficult and/or lengthy task due to the interaction between the transport phenomena and the kinetics that are present in this kind of systems [1].

In fact, the representation of the global rate of reaction is only possible when the simplicity of the kinetics allow it. In that case, the problem can be solved by employing analytical expressions. However, for reactions of industrial interest it is not common to represent the process with kinetics models as simple as the irreversible first order one. Therefore, analytical expressions to evaluate the global reaction rate are not available and numerical methods need to be used for most of the cases, for whose evaluation requires a lot of the computation time.

A great amount of time spent on numerical operations can be saved if a simplified method to evaluate the global rate of reaction were available. In this direction, some authors have obtained approximate solutions to the diffusion reaction problem with nonlinear kinetics. One way to simplify a model is by replacing a partial differential equation (PDE) for mass balance in the pellet by a proper ordinary differential equation (ODE). This idea is the basis of the methods proposed in References [2, 3, 4]. There are other ways to simplify the problem. Several authors [5, 6, 7, 8, 9, 10] have developed a method based on the Taylor series expansion for the reaction rate expression above the surface particle concentration for the key component. Their methodology has proven to be satisfactory for low values of the Thiele modulus, but the error grows when the Thiele modulus is increased due to the presence of the zero order term in the mass balance in the pellet as a consequence of the linealization process. The nonhomogeneous term can lead to spurious solutions, such as the presence of negative concentration values. To avoid this drawback, the proposed linear boundary-value problem is equipped with a nonactive region. Numerical results show that such a modification increases the prediction capacity of the analytical expressions.

II. THEORY

The average equation that governs the mass transport in a catalytic pellet for the isothermal case in transient systems is given by:

$$\frac{\partial U_A}{\partial \tau} = \frac{1}{\xi^m}\left[\frac{\partial}{\partial \xi}\left(\xi^m \frac{\partial U_A}{\partial \xi}\right)\right] - \Phi^2 \Re_A \tag{1}$$

Equation 1 is subjected to the following boundary conditions:

At the pellet center

$$\frac{\partial U_A}{\partial \xi} = 0 \qquad \text{at } \xi = \xi_c \qquad \text{for} \qquad \tau > 0 \tag{2}$$

At the pellet surface:

$$U_A = U_{in} \qquad \text{at } \xi = 1 \qquad \text{for} \qquad \tau > 0 \tag{3}$$

And the initial condition is:

$$U_A = U_0 \qquad \text{for } 0 \le \xi \le 1 \text{ when } \qquad \tau = 0 \tag{4}$$

In (1) – (4), U_A is the dimensionless concentration for the key component, Φ^2 is the Thiele modulus, ξ_c is the critical radius (position in the particle where the reactant becomes exhausted in the particle), and m indicates the geometrical shape parameter that takes the value: 0 for slab, 1 for cylindrical, and 2 for spherical geometry.

The boundary condition given by (2) is included in order to consider the case where the reactant concentration can become depleted at some intermediate position in the particle when the reaction rate is fast enough. On the other hand, critical radius

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 133–138.

is a function of the time [10, 11], but in this work we considered it a constant.

III. LINEALIZATION

To develop an analytic solution, we use the Taylor series to expand the reaction term for the surface concentration in the pellet:

$$\Re_A = \Re_A\big|_{\xi=1} + \left.\frac{\partial \Re_A}{\partial U_A}\right|_{\xi=1}\left(U_A - U_A\big|_{\xi=1}\right) \tag{5}$$

Substituting (5) in the mass balance, we can obtain the following linear PDE:

$$\frac{\partial U_A}{\partial \tau} = \frac{1}{\xi^m}\left[\frac{\partial}{\partial \xi}\left(\xi^m \frac{\partial U_A}{\partial \xi}\right)\right] - \beta - \gamma^2 U_A \tag{6}$$

where:

$$\beta = \Phi^2\left[\Re_A\big|_{\xi=1} - \left.\frac{\partial \Re_A}{\partial U_A}\right|_{\xi=1} U_A\big|_{\xi=1}\right] \tag{7}$$

$$\gamma^2 = \Phi^2 \left.\frac{\partial \Re_A}{\partial U_A}\right|_{\xi=1} \tag{8}$$

It should be noticed that β is a zero order reaction term, and if the approximate problem is solved, as a result of the presence of this term, negative values of the concentration might be obtained. It is clear that this situation will be predicted from the model at the starting of the reaction when the pellets are reagent depleted. To avoid this, a nonreaction zone is introduced in the analysis

IV. APPROXIMATE SOLUTION

After that the Laplace Transform has been applied, the solution for (6) with the boundary conditions given by (2) and (4) for the three geometrical shapes, given by *m*, are shown in Table I.

The constants and some of the details behind these equations are described in Appendix.

The averaged particle concentration is obtained by using [12]:

$$\left\langle U_A \right\rangle = (m+1)\int_{\xi_c}^{1} U_A \xi^m d\xi \tag{9}$$

In (9), we only considered the region where the reactant is present, which can be seen in the integration limits.

The average concentration was obtained by substituting the profiles concentration in (9). The results are shown in Table 2.

To find the critical radius, an additional boundary condition must be included, and it is given by

$$U_A = 0 \qquad \text{en } \xi = \xi_c \tag{10}$$

By substituting the Equations given in Table 1 in (10), we obtain an expression to calculate the critical radius. The results are presented in Table 3.

Table 1. Approximate concentration profile for the key component, for different pellets shapes: slab (*m*=0), cylinder (*m*=1) and sphere (*m*=2)

$m=0$	$U_A(\xi,\tau) = \frac{\beta}{\gamma^2}\left(\frac{Cosh\left[\gamma(\xi-\xi_c)\right]}{Cosh\left[\gamma(1-\xi_c)\right]} - 1\right) + \sum_{n=1}^{\infty}\frac{2Cos\left[\mu_n(\xi-\xi_c)\right]}{(1-\xi_c)Sen\left[\mu_n(1-\xi_c)\right]}e^{-\left(\mu_n^2+\gamma^2\right)\tau}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n\left(\mu_n^2+\gamma^2\right)} + \mu_n I_{in}\right]$	(11)
$m=1$	$U_A(\xi,\tau) = \left(U_0 + \frac{\beta}{\gamma^2}\right)\left(1 - \frac{\ln(\xi_c/\xi)}{\ln(\xi_c)}\right)e^{-\gamma^2\tau} + \frac{\beta}{\gamma^2}\left(\frac{I_0(\gamma\xi)K_1(\gamma\xi_c)+K_0(\gamma\xi)I_1(\gamma\xi_c)}{I_0(\gamma)K_1(\gamma\xi_c)+K_0(\gamma)I_1(\gamma\xi_c)} - 1\right)$ $+2\sum_{n=1}^{\infty}\frac{\left[Y_0(\mu_n\xi)J_1(\mu_n\xi_c) - J_0(\mu_n\xi)Y_1(\mu_n\xi_c)\right]e^{-\left(\mu_n^2+\gamma^2\right)\tau}}{B(\mu_n,\xi_c)}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n\left(\mu_n^2+\gamma^2\right)} + \mu_n I_{in}\right]$	(12)
$m=2$	$U_A(\xi,\tau) = \frac{\beta}{\gamma^2}\left(\frac{1}{\xi}\left\{\frac{\xi_c\gamma Cosh\left[\gamma(\xi-\xi_c)\right]+Senh\left[\gamma(\xi-\xi_c)\right]}{\xi_c\gamma Cosh\left[\gamma(1-\xi_c)\right]+Senh\left[\gamma(1-\xi_c)\right]}\right\} - 1\right)$ $+\frac{2}{\xi}\sum_{n=1}^{\infty}\frac{\left(\xi_c\mu_n Cos\left[\mu_n(\xi-\xi_c)\right]+Sen\left[\mu_n(\xi-\xi_c)\right]\right)}{Sen\left[\mu_n(1-\xi_c)\right]}e^{-\left(\mu_n^2+\gamma^2\right)\tau}\left[\frac{A_n}{D_n} + \frac{1}{C_n}I_{in}\right]$	(13)

Table 2. Approximate average concentration profile for the key component, for different pellets shapes: slab (m=0), cylinder (m=1) and sphere (m=2)

$m=0$	$\langle U_A\rangle=\frac{\beta}{\gamma^2}\left(\frac{Tanh\left[\gamma\left(1-\xi_c\right)\right]}{\gamma}+\xi_c-1\right)+\sum_{n=1}^{\infty}2\left(\xi_c-1+\frac{1}{\mu_n}\right)\frac{e^{-\left(\mu_n^2+\gamma^2\right)\tau}}{1-\xi_c}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n\left(\mu_n^2+\gamma^2\right)}+\mu_n I_{in}\right]$	(14)
$m=1$	$U_A\left(\xi,\tau\right)=-\left(U_0+\frac{\beta}{\gamma^2}\right)\left(\xi_c^2+\frac{1-\xi_c^2}{2\ln\left(\xi_c\right)}\right)e^{-\gamma^2\tau}+2\frac{\beta}{\gamma^2}\left(\frac{I_1(\gamma)K_1(\gamma\xi_c)-K_1(\gamma)I_1(\gamma\xi_c)}{\gamma\left[I_0(\gamma)K_1(\gamma\xi_c)+K_0(\gamma)I_1(\gamma\xi_c)\right]}-\left(1-\xi_c^2\right)\right)$ $+4\sum_{n=1}^{\infty}\frac{\left[J_1(\mu_n\xi_c)Y_1(\mu_n)-Y_1(\mu_n\xi_c)J_1(\mu_n)\right]e^{-\left(\mu_n^2+\gamma^2\right)\tau}}{B\left(\mu_n,\xi_c\right)}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n^2\left(\mu_n^2+\gamma^2\right)}+I_{in}\right]$	(15)
$m=2$	$\langle U_A\rangle=\frac{\beta}{\gamma^2}\left(\xi_c^3-1\right)+\frac{3\beta}{\gamma^4}\left(\frac{\left(\xi_c\gamma^2-1\right)Tanh\left[\gamma\left(1-\xi_c\right)\right]+\gamma\left(1-\xi_c\right)}{\xi_c\gamma+Tanh\left[\gamma\left(1-\xi_c\right)\right]}\right)+3\sum_{n=1}^{\infty}\left(\xi_c\mu_n{}^2+\frac{1}{\xi_c}\right)\frac{e^{-\left(\mu_n^2+\gamma^2\right)\tau}}{\mu_n^2}\left[\frac{A_n}{D_n}+\frac{2}{C_n}I_{in}\right]$	(16)

Table 3. Critical radius for different pellets shapes: slab (m=0), cylinder (m=1) and sphere (m=2)

$m=0$	$0=\frac{\beta}{\gamma^2}\left(\frac{1}{Cosh\left[\gamma\left(1-\xi_c\right)\right]}-1\right)+\sum_{n=1}^{\infty}\frac{2e^{-\left(\mu_n^2+\gamma^2\right)\tau}}{\left(1-\xi_c\right)Sen\left[\mu_n\left(1-\xi_c\right)\right]}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n\left(\mu_n^2+\gamma^2\right)}+\mu_n I_{in}\right]$	(17)
$m=1$	$0=\left(U_0+\frac{\beta}{\gamma^2}\right)e^{-\gamma^2\tau}+\frac{\beta}{\gamma^2}\left(\frac{1}{\gamma\xi_c\left[I_0(\gamma)K_1(\gamma\xi_c)+I_1(\gamma\xi_c)K_0(\gamma)\right]}-1\right)+\frac{4}{\pi\xi_c}\sum_{n=1}^{\infty}\frac{\mu_n}{B\left(\mu_n,\xi_c\right)}\left[\frac{\beta+U_0\left(\mu_n^2+\gamma^2\right)}{\mu_n^2\left(\mu_n^2+\gamma^2\right)}+I_{in}\right]$	(18)
$m=2$	$0=\frac{\beta}{\gamma^2}\left(\frac{\gamma}{\xi_c\gamma Cosh\left[\gamma\left(1-\xi_c\right)\right]+Senh\left[\gamma\left(1-\xi_c\right)\right]}-1\right)+\sum_{n=1}^{\infty}\frac{\mu_n}{Sen\left[\mu_n\left(1-\xi_c\right)\right]}e^{-\left(\mu_n^2+\gamma^2\right)\tau}\left[\frac{A_n}{D_n}+\frac{2}{C_n}I_{in}\right]$	(19)

V. INPUT FUNCTION

The factor I_{in} represents the contribution of the input concentration and it is given by:

$$I_{in}=\int_0^{\tau}U(\chi)e^{\left(\mu_n^2+\gamma^2\right)\chi}d\chi$$

It should be noticed that the effect of the type of input feed function is observed only in this term and the form of the solutions allows using any kind of input function concentration to the reactor. However, we considered the following input function:

$$U(\chi)=U\left(\tau-\tau_0\right)=\begin{cases}1 & \tau\geq\tau_0\\ 0 & \tau<\tau_0\end{cases}\qquad(20)$$

Therefore, in this case the term I_{in} is given by:

$$I_{in}=\frac{e^{\left(\mu_n^2+\gamma^2\right)\tau}-e^{\left(\mu_n^2+\gamma^2\right)\tau_0}}{\mu_n^2+\gamma^2}$$

In the following section we present the predictions of the approximate solutions and compare them with those obtained from the numerical solution of the nonlinear problem. In the sequel, the solution obtained with a highly accurate numerical method, finite differences in this paper will be referred as the *exact* solution. It should be mentioned that we have chosen the above input function only for the sake of illustration as it is representative of a class of input signals that are very likely to be found in practical situations.

VI. EXAMPLES

In this section in order to present a simple case of application and the corresponding results, the power law and Monod kinetics were considered.

A. POWER-LAW KINETICS

Considering a Power-law kinetics $\left(\Re_A=kU^n\right)$, the constants given by (7) y (8), are:

$$\beta=\Phi^2kU_s^n\left(1-n\right)\qquad(21)$$

$$\gamma^2=nk\Phi^2U_s^{n-1}\qquad(22)$$

In the Fig. 1, we present the average concentration profile, for this kinetic, considering the three geometrical systems used in this work.

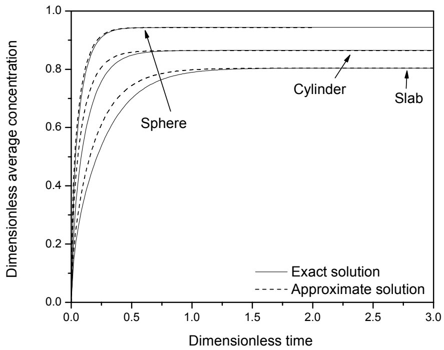


Fig. 1. Dynamic response of the particle average concentrations to a unit step input concentration. The parameters are: $n = 2$ and $\Phi = 1$.

B. MONOD KINETICS.

The Monod kinetic is given by:

$$\left(\Re_A = \frac{k_2 U}{1 + k_1 U} \right)$$

Therefore, in this case, the parameters are:

$$\beta = \frac{k_1 k_2 \Phi^2 U_s^2}{\left(1 + k_1 U_s\right)^2} \tag{23}$$

$$\gamma^2 = \frac{k_2 \Phi^2}{\left(1 + k_1 U_s\right)^2} \tag{24}$$

In Fig. 2 we present the average concentration profiles considering the Monod kinetic model

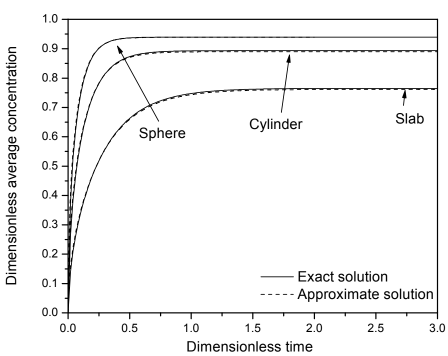


Fig. 2. Dynamic response of the particle average concentrations to a unit step input concentration. The parameters are: $k_1 = 0.01$ and $\Phi = 1$.

VII. DISCUSION

Given that the evaluation is based on the linealization of the reaction rate expression above the surface concentrations it is convenient to assess the differences with the exact values

Fig. 1 and 2 show the prediction for the approximate average concentration profile obtained from (14) – (16). The results are compared to those obtained from the numerical solution of the nonlinear diffusion-reaction problem.

In Fig. 1 it is observed that there is a slight difference when the time is low enough for the second order reaction (n =2). In this case, the error percent is less than 10% for the slab geometry, which is an acceptable deviation. Nevertheless it should be noticed that in steady state, both solutions are the same. However, the error grows when the Thiele modulus is increased ($\Phi^2 > 10$), as consequence of the order zero term in the mass balance.

On the other hand, Fig. 2 shows the dynamic behavior of the reactor with the Monod kinetic. In this case, both the approximate and exact solutions are the same. This characteristic is due the parameter value used: $k_1 = 0.01$, because in this case the Monod kinetic is similar to the first order one [13], and the problem is lineal. However, the error grows to intermediate values of this parameter ($1 \le k_1 \le 10$). Although, we do not present the graphics, for larger values the behavior of the approximate solutions are equals than exact solution, due the other extreme case of the Monod kinetic: the order zero reaction [13].

Summing up, the approximate solution from the proposed method has the advantage that computations are drastically reduced as compared with strict numerical methods (e.g., finite elements and finite differences). This feature makes the proposed method quite suitable for process evaluation where exhaustive simulations can be required.

We have concluded after exhaustive simulations (the ones shown in the above section are only a small but representative part of them) that the applicability of the approximate method is restricted to situations with small or moderate Thiele modulus. Relatively high Thiele modulus values, larger than around 10, induce sharp particle concentration profiles, which generate large approximation errors.

VIII. CONCLUSIONS

An approximate method to evaluate global reaction rate has been developed, as a tool to reduce the calculations needed in a reactor simulation. The method is applicable to any kinetic model equation. The comparison of the concentration profile with the ones from the exact solution shows that the method can be used for low values of the Thiele modulus, but we avoid to find negative values for the concentration introducing a critical radius.

NOTATION

U_A	Dimensionless concentration for the key component
τ	Dimensionless time
ξ	Dimensionless radius
m	Geometrical shape parameter (0 for slab, 1 for cylindrical, and 2 for spherical geometry)
Φ^2	Thiele modulus
$\Re_A$	Dimensionless reaction rate
ξ_c	Dimensionless critical radius
U_0	Dimensionless initial concentration
U_s	Dimensionless surface concentration
$\langle U \rangle$	Dimensionless average concentration

subscripts

A	Key component.
s	Pellet surface.

REFERENCES

[1] J. O. Marroquín de la Rosa, J. A. Ochoa Tapia, T. Viveros García, "Método de estimación aproximado del factor de efectividad isotérmico aplicable a cualquier cinética", Avances en Ingeniería Química, 8 (1), 37-43, (1998).

[2] M. Goto and T. Hirose , "Approximate rate equation for intraparticle diffusion with or without reaction", Chemical Engineering Science, 48(10), 1912-1915, 1993.

[3] M. K. Szukiewics, "New approximate model for diffusion and reaction in a porous catalyst", AIChE Journal, Vol. 46 (3), 661-665, 2000.

[4] M. K. Szukiewics, "Approximate model for diffusion and reaction in a porous pellet and an effectiveness factor", Chemical Enginnering Science, 59, 479-483, 2004.

[5] H. W. Haynes, "An explicit approximation for the effectiveness factor in porous heterogeneous catalysis", Chemical Enginnering Science, 41, 412-415, 1986.

[6] J. O. Marroquín de la Rosa, J. A. Ochoa Tapia, T. Viveros García, "A linear approximation method to evaluate isothermal effectiveness factors", Chemical Engineering Community, Vol. 174, pp. 53-60, 1999.

[7] M. K. Szukiewicz, "An approximate model for diffusion and reaction in a porous pellet", Chemical Engineering Science 57, 1451-1457, 2002.

[8] J. O. Marroquín de la Rosa, J. A. Ochoa Tapia, T. Viveros García, "Approximate isothermal global effectiveness factor", Revista Mexicana de Ingenieria Quimica, Vol 2, 183-191, 2003.

[9] J. A. Ochoa Tapia, F. J. Valdes Parada and J. J. Alvarez Ramirez, "Short-cut method for the estimation of isothermal effectiveness factors", Industrial Engineering Chemical Resource, Vol. 44, pp. 3947-3953, 2005.

[10] F. J. Valdes Parada, J. J. Alvarez Ramirez and J. A. Ochoa Tapia, "An approximate solution for a transient two-phase stirred tank bioreactor with nonlinear kinetics ", Biotechnology Progress, Vol. 21, pp. 1420-1428, 2005.

[11] C. A. Gonzalez Rugerio and J. A. Ochoa Tapia, "Dynamic behavior of a continuous stirred tank reactor of two phases with a order-zero kinetics", Revista Mexicana de Ingenieria Quimica, Unpublished.

[12] S. Whitaker, "The method of the Volume Averaging". Kluwer Academic Publishers, (1999).

[13] J. J. Carberry, Chemical and catalytic reaction engineering, Dover publications, Inc. EUA, 2001.

APPENDIX

In this section we present the details of the solutions obtained in this work.

Slab $(m=0)$

In this case, (1) is given by:

$$\frac{\partial U_A}{\partial \tau} = \frac{\partial}{\partial \xi}\left(\frac{\partial U_A}{\partial \xi}\right) - \beta - \gamma^2 U_A$$

The solution of the problem is accomplished using the Laplace Transform method.

$$\bar{U} = \frac{1}{\lambda^2}\left(\bar{U}_{in}\lambda^2 - U_0 + \frac{\beta}{s}\right)\frac{Cosh\left[\lambda(\xi-\xi_c)\right]}{Cosh\left[\lambda(1-\xi_c)\right]} + \frac{U_0}{\lambda^2} - \frac{\beta}{\lambda^2 s} \quad \text{(A1)}$$

where:

$$\lambda^2 = s + \gamma^2$$

At this point the solution of the problem has been obtained in the Laplace domain. The concentration of the particle is obtained by the inversion of (A1) and the solution is given by the following expression:

$$U(\xi,\tau) = \frac{\beta}{\gamma^2}\left(\frac{Cosh\left[\gamma(\xi-\xi_c)\right]}{Cosh\left[\gamma(1-\xi_c)\right]} - 1\right) + \sum_{n=1}^{\infty}\frac{2Cos\left[\mu_n(\xi-\xi_c)\right]}{(1-\xi_c)Sen\left[\mu_n(1-\xi_c)\right]}e^{-(\mu_n^2+\gamma^2)\tau}\left[\frac{\beta+U_0(\mu_n^2+\gamma^2)}{\mu_n(\mu_n^2+\gamma^2)} + \mu_n I_{in}\right] \quad \text{(A2)}$$

where μ_n is given by:

$$\mu_n = \frac{(2n-1)\pi}{2(1-\xi_c)};\ n = 1,2,3... \quad \text{(A3)}$$

Cylinder $(m=1)$

For this case, the equation that governs the process is:

$$\frac{\partial U_A}{\partial \tau} = \frac{1}{\xi}\left[\frac{\partial}{\partial \xi}\left(\xi\frac{\partial U_A}{\partial \xi}\right)\right] - \beta - \gamma^2 U_A \quad \text{(B1)}$$

The application of the Laplace Transform operator yields a boundary value problem in the Laplace domain. The solution of this problem is:

$$\bar{U} = \frac{U_0}{\lambda^2} - \frac{\beta}{\lambda^2 s} + \frac{1}{\lambda^2}\left(\bar{U}_{in}\lambda^2 - U_0 + \frac{\beta}{s}\right)\frac{I_0(\lambda\xi)K_1(\lambda\xi_c) + K_0(\lambda\xi)I_1(\lambda\xi_c)}{I_0(\lambda)K_1(\lambda\xi_c) + K_0(\lambda)I_1(\lambda\xi_c)} \quad \text{(B2)}$$

The inverse of this equation is given by (12), and μ_n is calculated by:

$$J_0(\mu_n)Y_1(\mu_n\xi_c) - J_1(\mu_n\xi_c)Y_0(\mu_n) = 0 \quad \text{(B3)}$$

and the constant included in (12), (15) and (18) is:

$$B(\mu_n,\xi_c)=J_1(\mu_n\xi_c)Y_1(\mu_n)-J_1(\mu_n)Y_1(\mu_n\xi_c)$$
$$+\xi_c\left[J_2(\mu_n\xi_c)Y_0(\mu_n)-J_0(\mu_n)Y_2(\mu_n\xi_c)\right] \tag{B4}$$

Sphere $(m=2)$

In this case, the mass balance can be described by:

$$\frac{\partial U_A}{\partial\tau}=\frac{1}{\xi^2}\left[\frac{\partial}{\partial\xi}\left(\xi^2\frac{\partial U_A}{\partial\xi}\right)\right]-\beta-\gamma^2U_A \tag{C1}$$

The solution in the Laplace domain is:

$$\bar{U}=\frac{U_0}{\lambda^2}-\frac{\beta}{s\lambda^2}$$
$$+\frac{1}{\xi}\left(\bar{U}_s-\frac{U_0}{\lambda^2}+\frac{\beta}{s\lambda^2}\right)\left[\frac{\lambda\xi_c Cosh\left[\lambda(\xi-\xi_c)\right]+Sinh\left[\lambda(\xi-\xi_c)\right]}{\lambda\xi_c Cosh\left[\lambda(1-\xi_c)\right]+Sinh\left[\lambda(1-\xi_c)\right]}\right] \tag{C2}$$

The inverse is given by the equation (13), and the constants are:

$$A_n=2\left[\beta+U_0\left(\mu_n^2+\gamma^2\right)\right]$$
$$C_n=\xi_c(1-\xi_c)+\frac{1}{\mu_n^2\xi_c} \tag{C3}$$
$$D_n=\left(\mu_n^2+\gamma^2\right)\left[\mu_n^2\xi_c(1-\xi_c)+\frac{1}{\xi_c}\right]$$

And μ_n are the roots of the following equation:

$$\xi_c Cos\left[\mu_n(1-\xi_c)\right]+\frac{Sen\left[\mu_n(1-\xi_c)\right]}{\mu_n}=0 \tag{C4}$$

DESCRIBING ACOUSTIC FINGERPRINT TECHNOLOGY INTEGRATION FOR AUDIO MONITORING SYSTEMS

Carlos Serrão
carlos.serrao@iscte.pt
ADETTI/ISCTE
ISCTE - Instituto Superior das Ciências do Trabalho e da Empresa
1600-082 Lisboa, Portugal

Marco Clara
antunes.clara@marinha.pt
Direcção de Tecnologias de Informação e Comunicação
Marinha Portuguesa
1149-001 Lisboa, Portugal

Abstract - **The ability to create, write or compose a song is a gift or feature that not everyone has within him or herself. The results of that kind of ability are an asset that should somehow, due to their nature, be recorded and protected. That protection can be enabled by the enforcement of the author's rights, safeguarding its works. How can this protection be enforced in a digital World? Should we trust that nobody steals or circumvents intellectual or artistic property? Or should we think of possible ways to control and monitor that property's usage? It seems that the last one is currently the right way, and technologies such as acoustic fingerprint may allow us to provide such monitoring and enforcement. In what way can we integrate existing technology for creating such systems, and what criteria should be used for evaluating that same technology?**

Index Terms - **DRM, Digital, Rights, Restrictions, Management, Audio, Acoustic, Fingerprinting**

I. INTRODUCTION

Music is practically a constant in our day-to-day life. Imagine this possible following daily scenario: we wake up in the morning, and listen to the radio station that is playing our wake up song. Some of us even use a small radio in the bathroom so that we can hear it while taking a bath. We drive our cars to the job, still listening to our favourite radio, our favourite CD or a MP3 compilation of our favourite songs. We arrive at work, and plug in our headphones so we can keep listening to music. We go to lunch and the restaurant is playing ambience music. We get back to work and to our headphones. We go back home, driving our car and still listening to music. We arrive home and turn on our audio system or start watching a concert in our DVD player. If we go out to a bar, or a disco, music is still there. In practice we can stay all day long consuming music from several different sources and not even thinking about it – it is something that has become "natural" in our lives. We may not even be asking for it, and yet, it's there. So what about the music creators/authors? Who's taking care of their interests or their rights? If we buy a CD or a DVD (disregarding copy protection issues) their rights are safeguarded, but what about all the other situations? The resulting need for systems that can enforce some type of protection and monitoring is becoming more and more important and several technologies are now emerging so that these systems can be implemented. We will start by giving some ideas about some of the most common systems used.

Digital Rights Management (DRM) [1] is the concept that refers to a set of several different technologies that allow enforcing control policies, in what concerns digital data (software, hardware), video and audio. The monitoring of the rights, also enclosed within DRM concept, is probably one of the most difficult tasks to enforce, and it deserves some attention.

However, one should not be confused about this concept (DRM) and others such as "copy protection". In this case we are talking about technologies that simply allow controlling the access, or restricting the usage of digital media, according to the used device for playing it, for example. Some critics claim that a more accurate designation for DRM should be Digital "Restrictions" Management [16]. But, if we start using the "restrictions" term, we are somehow limiting DRM concept, in the sense that, for example, the usage monitoring of contents for rights recording is also part of DRM enforcing. Anyway, the main use of DRM related technologies is considered to be the revenue loss prevention due to illegal duplication of copyrighted contents [17], although this is not the main issue for this document.

According to the previously mention needs concerning enforcement of DRM policies, and specifically, to the monitoring ability of content usage, it is important to clarify some aspects related to a specific technology. This paper is focused primarily on audio content monitoring and protection, the acoustic fingerprint (also called audio fingerprint).

The main objective of an acoustic fingerprint mechanism is to efficiently and robustly compare the equality (or not) of two audio files, not by comparing the files themselves, but by comparing substantially smaller sets of information, referred to as acoustic fingerprints [8]. Usually, systems relying on acoustic fingerprinting, all the audio file information, including the file itself, artist, title, album duration or any other kind of related information is maintained in some kind of database, being the fingerprint the main identifier (or index) for the record. This way, when searching a record for comparison, one may obtain more quickly and robustly a result, either if there is a match or not in the database. In result, the fundamental aspects of this technology can be identified:

- Fingerprint generation mechanism
- The fingerprint itself (previously generated)
- The fingerprint matching mechanism
- Database for audio related information (including the fingerprint as index)

This paper will describe in further detail the acoustic fingerprinting technology, not only by covering the previously mentioned fundamental aspects or components, but also considering integration aspects related to usage of this technology on other information systems. This way, we intend to provide some information that can be used as a set of guidelines for analysing related technology and

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 139–144.

implementation mechanisms, regarding audio monitoring systems.

II. TECHNOLOGY

A. Description

An acoustic fingerprint can be considered as some kind of DNA (deoxyribonucleic acid) scheme for the associated audio file or object. This way we can say that, like DNA, the acoustic fingerprint of any audio file will be unique. We can also associate the concept of hash functions to acoustic fingerprinting, since a hash function F should map a large object X to a smaller corresponding hash value. So, we can think about comparing hash values for two submitted objects for the same hash function. This way, if the obtained hash value is correspondent, then the submitted objects should in fact be the same. The only difference between a simple hash function and an acoustic fingerprinting algorithm is that when we think of the objects they may not in fact be the same. Today's audio formats allow us to possess the same audio track with different features, in what concerns, for example, bitrate or other aspects regarding audio quality. So, this means that an acoustic fingerprinting algorithm must be based on the perceptual features of an audio object, and not the "physical" features of the object itself (for example, binary code), feature that obviously falls out of a simple hash function scope.

Despite the mentioned perceptual features of the audio files to be analysed, one should distinguish between a song with varying quality, and two versions of the same song (for example, sung by artist A and sung by artist B). In this case it can be said that such kind of correspondence is virtually impossible to attain. Even though some work is being done in what concerns constructing a "fingerprint function in such a way that perceptual similar audio objects result in similar fingerprints" [8], as described by Jaap Haitsma and Ton Kalker, from Philips Research.

One of the aspects related to an acoustic fingerprinting algorithm is its robustness. We will see in the Open Fingerprint Architecture section (see Section III.B) that to obtain the right robustness, the algorithm should be based on certain perceptual features of the audio, so that even with degradation (radio transmission, for example) the audio signal should still be correctly identified. This is possible through the analysis of invariant features of that same audio that will lead (hopefully) to generating the same fingerprint for different quality (but same audio) content. Haitsma and Kalker [8] also refer to aspects such as "reliability" and "granularity" as fundamental, but as you will see these aspects are indeed mainly related to the robustness of the acoustic fingerprinting algorithm itself. As for the "fingerprint size", "speed" and "scalability" issue, it is all a matter of resources and coding. Since the fingerprint data itself is not that large, storage is practically not a problem. This way, robustness is in fact the main issue when it comes to acoustic fingerprinting, even though all the other mentioned aspects should still be considered very important.

B. Client Applications

One of the most desired uses for acoustic fingerprinting, like mentioned before, is the creation of equally robust monitoring mechanisms that can somehow allow the enforcement of some DRM principles. A simple target of such mechanisms is most definitely the **radio stations broadcasting**, that could this way be precisely monitored, registering every single song played during a certain period. That would allow a more effective and efficient digital rights management and application. Actually, in several countries the best that rights management societies can do is some kind of statistic treatment of the data related to these broadcasts, using samples to calculate which songs are played, with low degree of accuracy. This makes obvious all the advantages that the monitoring systems would bring to this area, giving a more precise feedback regarding multiple channel audio broadcasting. Through this mechanism there would be automatically generated playlist reports for a simpler royalty collection, instead of the mentioned manual process of determining which songs are more or less played in that radio or web station, for example.

Further in this paper it is possible to understand the server technology regarding audio fingerprinting services and data storage, but what about **monitoring systems**? They are one of the perfect candidates to a correctly elaborated plan in what concerns the design of client applications. One can think of a system such as this, as a separate part of the server's architecture. They can take form, for example, of some kind of intermediate appliance between the fingerprint server and the transmission in course. A monitoring appliance could this way be "listening" to a broadcast, registering all the played songs, and automatically identifying and recording those same songs in a log for further future analysis. This way we open the door to a simple and integrated implementation recurring to existent systems and data (or metadata) repositories, making use of (in a perfect world) large sets of information for music identification.

However, we do not live in such a perfect world, and the radio station broadcasting is also the perfect example for the difficulties associated with monitoring systems: the **audio quality**. This issue is a sensitive one, since it makes necessary to have a very robust audio fingerprinting mechanism that can support all the quality degradation present on an analogical signal radio transmission (just think of the difference between the quality associated with AM and FM radio stations and frequencies).

C. Server Applications

It is fair enough to say that, beside algorithms, techniques related with fingerprint generation or client applications of any kind, the remaining complexity is fairly attributed to server side technology. It is possible to understand that, due to the complexity and purposes of a server that delivers data or metadata related to music or song records, more precisely to scalability details, performance requirements, storage capability, etc., this is a crucial point of any adopting architecture. MusicIP [5] is one of the currently greatest metadata and acoustic fingerprint providers in the market, and is also the owner of the repository containing one of the largest sets of information available. This organization is described in further detail on the next section. According to a recent announcement from MusicIP, "the company's global music search database has reached the milestone of 20.000.000 track IDs and musical signatures" [9]. Those tracks are identified by the company's patented fingerprint information and also associated with music "algorithmic analysis" [9], for song characterization. Since March 2006, when there was a total of 16 million records to June 2006, when it has been reach the previously mention 20 million records, it was inserted a total of 4 million tracks, resulting in something like 42000 tracks inserted per day. All of this information, considered as the largest existing database, is available through MusicIP's MusicDNS [6] (Musical Digital Naming Service). Basically this server-side service is available as a web service, and allows track identification and associated metadata retrieval. Currently MusicIP is aiming to be supplied in what concerns music information, from artists all over the world. In this sense MusicIP provides a global infrastructure, more precisely in the form of a Global Music Search Engine [9] with a background scalable framework.

D. Client-Server Communication

Once the client and server parts of an acoustic fingerprint based system are specified, and once it is understood how each one individually works and what are their functions, it is necessary to understand how they communicate or can communicate with each other.

This communication may occur in several ways. As mentioned previously, on the server application specification, one of the most common is through a web service, exchanging information through a standard format such as XML [10]. One of the technologies also mentioned at the next section of this document, more precisely Open Fingerprint Architecture [6], also uses this method for exchanging information. More accurately, XML is used when sending information from the server to the client, while a simple HTTP request is sent from the client to the server while the request for data (or metadata) is being made. Basically, the currently available HTTP based API and protocol specification [11] allows sending requests for three kinds of track lookups:

- By fingerprint (previously generated)
- By PUID (Portable Unique ID of the track)
- By SHA-1 signature (file based or acoustic based)

This request is made by sending an HTTP form using the POST method, to the MusicDNS track lookup service, concatenated with one or more arguments, obtaining as a result an XML document generated using the provided arguments. A list of possible parameters is presented on the following table:

Param.	Description
cid	Client ID that can be obtained at http://ofa.musicdns.org
cvr	Version of the client
fpt	Acoustic fingerprint as generated by open source client
uid	Portable Unique ID, for track lookup by PUID
s1f	SHA-1 hash value, file based
s1a	SHA-1 hash value, acoustic based
rmd	Flag to return metadata (0 or none for returning only the PUID, 1 for returning available metadata on the resulting XML document)

The service also requires another set of information, available before the lookup, which will be used for correcting errors and creating more data for the future:

Param.	Description
brt	Bitrate of original sample in kbps
fmt	Audio format (MP3, WMA, Ogg, AAC…)
dur	Track length in milliseconds
art	Artist name
ttl	Song title
alb	Album name
tnm	Track number
gnr	Music genre
yrr	Release year
cmp	Composer of the song
cnd	Conductor of the song
lyr	Lyricist of the song
orc	Orchestra of the song
enc	Encoding type (related to text data, such as UTF-8 by default, not audio format)

Several of these parameters are optional and, if not specified, assume default values as 0 or "unknown". The required parameters vary according to the type of lookup that is being done. The following parameters are required, for each specified lookup type.

<u>Fingerprint lookup:</u>
cid cvr + s1f s1a rmd cmp cnp lyr orc enc
<u>PUID lookup:</u>
cid cvr + s1f s1a rmd brt fmt dur cmp cnd lyr orc enc
<u>SHA-1 file based lookup:</u>
cid cvr + s1f rmd brt fmt dur art ttl alb tnm gnr yrr cmp cnd lyr orc enc
<u>SHA-1 acoustic based lookup:</u>
cid cvr + s1a rmd brt fmt dur gnr cmp cnd lyr orc enc

Notice that the SHA-1 file and acoustic based lookup parameter list have been corrected from the original document [11], since the *s1f* and *s1a* arguments where swapped between formulas.

The resulting XML document, supported by the current MusicBrainz schema [12], responding to the elaborated request, should look like this:

```
<?xml version="1.0" encoding="UTF8"?>
<metadata xmlns="http://musicbrainz.org/ns/mmd/1/">
	<track>
		<title>xxx</title>
		<artist>
			<name>yyy</name>
		</artist>
		<puidlist>
			<puid id="zzz"/>
		</puidlist>
	</track>
</metadata>
```

In case the *rmd* parameter is sent with value 0 or default, then the resulting XML would only contain the PUID list inside the track data, with no other metadata included.

III. PRACTICAL CASES

Currently, some work is being developed by a few organizations, not only commercially but also in open source communities, related to the usage of acoustic fingerprinting technology. Enclosed within the open source concept related to acoustic fingerprinting, some of these projects even allow downloading some client and server software (and source code), and even their database previously loaded with audio information (including fingerprint information). MusicBrainz [2] and Open Fingerprint Architecture [3] are some of the most important projects currently available, related with the application of acoustic fingerprinting.

A. MusicBrainz

MusicBrainz [2] consists in an open source project, composed by a community music meta-database, associated to a music information site. MusicBrainz data can be browsed through its website or through other client program. A CD player can use MusicBrainz to identify media, providing information about the album, artist or other related information. With MusicBrainz Tagger one can also identify and clean up metadata tags in musical collections. There can also be made contributions to this project, by adding information to the database, while using the tagger software. Some of the database information is within Public Domain, while other is enclosed by Creative Commons Licence. Note that MusicBrainz provides information about audio recordings, but not the audio files itself.

MusicBrainz catalogue can be used for browsing and searching songs, for any registered user. Within this catalogue is contained metadata that allows to uniquely identifying music, either through a direct search requested by a user or through an information lookup performed by a media player.

This project makes use of acoustic (or audio) fingerprinting features, using this method to generate unique identifiers for each song registered in the metadatabase. Up to this moment it has been using Relatable's TRM [4], however the obtained results are not considered satisfactory (poor scalability, poor performance). Since this solution is not open source, the detected bugs and problems are not being corrected, so, since March 2006, MusicBrainz is collaborating with MusicIP [5], so that their system, designated by MusicDNS [6], can be used. At first it was considered by the team to develop their own fingerprinting solution, but they have preferred to use this system, even thought it is still proprietary.

Relatable's TRM acoustic fingerprinting system makes use of music IDs representing audio signatures for musical content. The tagger application sends data from a song to the server, so that an ID can be generated. That ID can then be used to identify the song through the database, and retrieving metadata for that song. This system does not allow identifying a song through fuzzy matching (a song that sounds similar to other song).

Regarding MusicIP's technology, PUID's identifiers and Acoustic Fingerprints are not the same. PUID is simply a unique identifier, used by MusicDNS, associated to music information. This unique identifier is used for music analysis, and requires the track to be fully analyzed (takes up to 80% of the track's time). This analysis can then be sent to MusicDNS's server and used for fuzzy matching. Notice that this technology is not open source, so it cannot be included or integrated within taggers or other clients enclosed within GPL licence or similar situations. Acoustic fingerprint is a lighter process regarding music identification, since it can analyse a smaller portion of the track, calculating the audio fingerprint (in this case through the open source "*libofa*" library [3]). With the resulting fingerprint a lookup can be done through MusicDNS web-service, retrieving, if there is a match, the corresponding PUID, previously submitted through the music analysis process, mentioned before. Fingerprint lookup service is free, for MusicBrainz related projects and other open source projects. The *libofa* process does not allow submitting new PUIDs, since this mechanism only contains enough information for generating PUIDs suitable for matching.

MusicBrainz also delivers an XML web service, as described in detail on the previous section, which can be used as an interface to the database, containing data maintained by the community. This way, access to music metadata is available for software (existing music players, etc.) and developers (for designing new client applications). Contents are delivered to those, by a simple and flexible XML format.

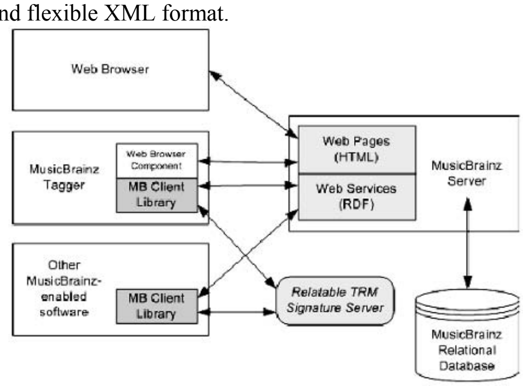


Fig. 1 – MusicBrainz Service Architecture

Notice that the available web service is the core mechanism for establishing communication with client applications such as the MB Tagger. HTML content is also delivered by the MusicBrainz's server component.

B. Open Fingerprint Architecture

Open Fingerprint is an architecture designed for use with digital music [3]. It was designed for music recognition and not voice or sound sample recognition. Its goal is to uniquely identify individual songs or tracks and it is not built to work with MIDI, samples or other kind of files.

This technology is format independent and is able to deal with several different formats, such as MP3, AAC, Ogg, WMA, FLAC, etc. Open Fingerprint is also bitrate and loss independent, meaning it can deal with the lowest tolerable bitrate (down to 65 kbps) to the highest bitrate available, and supporting "lossy" formats such as the well known MP3 to "lossless" formats as WAV or FLAC. It does not deal with issues outside of music recognition scope, meaning it as no integration whatsoever with legal issues or DRM enforcement. It simply identifies music.

Currently there are several acoustic fingerprinting solutions available on the market. Mainly their objective is to efficiently and effectively identify songs through a small portion of it. A few years ago in Portugal, a mobile operator turned available a service that allowed, by holding up your phone to a speaker playing a song, just for a few seconds, to identify that same song. There are some obvious problems related to this kind of technology, such as the difficulty to identify a song through a very small portion of it (comparing to the total track length), and even in what concerns the monitoring issues, since it is easier to "mislead" the monitoring system by simply adding portions of public domain content so that it can be fooled, by not detecting the protected content.

According to acoustic fingerprinting terms, there are two types of mistakes that can happen: the false positive (when a song is incorrectly identified) and the false negative (when a song cannot be identified). According to Open Fingerprint specifications, the used algorithms are designed and tuned up to avoid these kinds of errors.

The OFA is designed to deal with a very large set of information (in a perfect world, eventually with the whole world's music), opposing to some of the acoustic fingerprinting systems mentioned before (those who work better with smaller sets of popular music, easily identified with captured small fragments).

The process of generating an Open Fingerprint is composed by a few steps, aiming to obtain a 516 byte array that can be sent to the MusicDNS server, for identification and song ID retrieval. However there are some setbacks in what concerns how this process can occur.

Audio decoding is considered unavoidable, so that fingerprinting can take place. There is also the overhead issue, since this process, even though it's fast (2 minutes or less depending on the song's length) and light (less than 2000 code lines, according to MusicIP), has to deal with the decoding issue. The "2 minute" issue is a way of making identification "safer", since, although is possible to have a 2 minute "audio garbage" in front of a song, probably nobody would tolerate it. This way, the small fragment vulnerability is overcome.

After decoding is complete, the fingerprinting process starts normalizing the signal. This way it can be transformed according to a standard set of properties. Open Fingerprint uses FFT, standing for Fast Fourier Transform [7], that "in signal processing and related fields (…) is typically thought of as decomposing a signal into its component frequencies and their amplitudes" [1]. FFTW is a library release available under GPL. Intel Math Kernel Library is a low cost library, also for FFT. One of these libraries is required for using the MusicDNS Open Fingerprint service, even though the usage of other libraries may be considered.

Once the normalization process is completed, the returned amplitude data is examined. The series of spectra containing small frames of 185 milliseconds of audio are then transformed into a matrix of time versus frequency. Once this matrix is available, it is then applied the so called SVD, or Single Value Decomposition [3] equation, which states:

"*For an mbyn matrix A with m >= n, the singular value decomposition is an mbyn orthogonal matrix U, an nbyn diagonal matrix S, and an nbyn orthogonal matrix V so that A = U*S*V'. The singular values, sigma[k] = S[k][k], are ordered so that sigma[0] >= sigma[1] >= ... >= sigma[n1]*"

The result obtained through this equation is a much smaller matrix than the original one, more exactly a 516 byte matrix that is much lighter for later use (and probably transmission across a network).

The final phase of the process regards obtaining or capturing peaks, revealing continuities between frames. These peaks are cumulatively measured so that a ranking of strong pitches can be created. The top four strongest pitches are then transformed into bytes, and used for narrowing fingerprint matching.

Regarding data storage, MusicIP claims that there is no mathematical danger of running out of space. MusicDNS server farm is built over Predixis fingerprint servers, and uses distributed query n-tier architecture, similar to some search engines such as Google. Performance is optimized maintaining core information in RAM, which in conjunction with the SVD comparison, allows about 100 simultaneous fingerprint lookups per second.

All previous steps mentioned before aim to obtain a unique ID, which in the end is provided along with the rest of the analysed track's metadata.

C. Other Projects

The previous inter-related projects where mentioned as they are probably the most "cutting edge" currently taking place. Although, working over acoustic fingerprinting technology is a common effort, and other projects are currently taking place, either in a commercial and non-commercial perspective. In the open source area, other projects such as MusicURI can be mentioned.

MusicURI [13] is an ongoing research, which aims to develop an "infrastructure that enables the mapping of digitally encoded music resources against unique, consistent or seldom modifiable Universal Resource Identifiers (URI) by building upon existing Semantic Web and multimedia technologies". As it has been already explained in detail, acoustic fingerprinting allows identifying audio, regardless of signal degradation due to transmission, compression, filtering, etc. MusicURI uses the MPEG-7 [14] standard that defines a universal mechanism for exchanging data associated with multimedia content. It also has Audio Signature Description Scheme as its acoustic fingerprinting tool, for robust audio identification.

It is important to have in mind that currently there are no definitive standards to adopt, in what concerns acoustic fingerprinting and related technologies available. The mentioned projects are important to give us that very same perspective, so that somehow, an independent analysis of the main aspects can be done regarding integration issues to be considered.

IV. IMPLEMENTATION

Some of the mentioned projects through the previous sections, as also described, allow downloading open source client and library code. This way one can understand the main steps already described, regarding the fingerprint generation through audio analysis.

A. OFA Library 0.9.3

In this case it will be used, for example purposes, the Open Fingerprint Architecture library, currently available at MusicDNS website, under the following designation of ***libofa-0.9.3***. When exploring the lib folder of the zip file containing the library, it can be found the C++ code for fingerprint generation.

B. Songprint 1.2

There is also another library fairly used, for example by client application MPT (Mike's Playlist Thingie) [15], which is particularly built to deal with playlists (and which code is also available as open source). This library is called Songprint. As you can imagine, Songprint is also a library to identify music based on what it sounds like. Mike's "Thingie" (at least at it's current version) recommends the usage of Songprint 1.1, but the most current version of the library is 1.2 nowadays, and can be found by the designation ***songprint-1.2***.

This library is built in C language, and as, such as *libofa*, a fairly simple structure. Under the *src* folder one can find the main classes of the library, including a "special treat" regarding codecs for allowing the ability to deal not only with uncompressed, but also compressed formats (*mpg*, *mp3*, *vorbis*…).

It is fair to say that in either case (*libofa* and *songprint*) the used code corresponds to a similar structure, in what concerns the transmitted information over the network or even the used mechanisms for audio analysis and pre-processing, for subsequent signature generation. This way it is even possible to consider, for example, creating an abstraction layer that may somehow allow using several libraries through a simple auxiliary configuration that maps all the associated methods to each library used. Even the related coding issues may be separated or aggregated to a mechanism of this kind.

V. CONCLUSIONS

As it was presented in this paper, there are currently several efforts taking place for attaining the same set of goals, on what concerns the acoustic fingerprinting technology and its availability. It important to consider the aspects related to the context in which these efforts take place. One of these aspects refers to the nature of the technology exploitation either in a more commercial oriented way or if they are part of a common effort, within a collective scope such as the open source community. It is also important to consider that this open-source development context can somehow ease further development and even standardization in this area.

The information domain issue should also continue to be, somehow, debated or even negotiated. It is a common good to systems that make use of acoustic fingerprint, or that rely on musical information and data, to have access to large sets of information within the public domain scope. Since the storage of this kind of information is generally, as it was demonstrated, based on metadata associated to a musical content, and not the content itself, any problem regarding this issue should easily be overcome.

If these and other "satellite" issues regarding acoustic fingerprinting can be solved, it will be easier to start developing systems that integrate the existing technologies, that even though are a mixture of proprietary code and information with open source (mainly client) code and some public information, can be considered to be in the right way. At least and as it was described in the integration section, it seems very obvious that any development team could easily start using bits and pieces of the described technologies to develop their own system, and even start running their own test servers.

Some libraries where analysed within the structure of this document, at a code-level based detail, and the possibility of integrating that very same code, and reusing it, associated to the corresponding server technology, will allow designing new acoustic fingerprint based systems that can benefit with the currently ongoing growth of stored information, more precisely metadata associated to musical content. MusicBrainz database and eTantrum servers are good examples of the already large, but still growing, sets of information currently available.

REFERENCES

[1] Wikipedia website, <http://www.wikipedia.org>
[2] MusicBrainz website, <http://musicbrainz.org>
[3] Several authors, "Open Fingerprint Architecture Whitepaper", Music IP 2006
[4] Relatable website, <http://www.relatable.com>
[5] MusicIP website, <http://www.musicip.com>
[6] MusicDNS website, <http://www.musicdns.org>
[7] Fast Fourier Transform in the West website, <http://www.fftw.org>
[8] Jaap Haitsma and Ton Kalker, "A Highly Robust Fingerprinting System", IRCAM Centre Pompideu 2002
[9] Several authors, "Largest Music Search Collection Gets Even Larger", June 21st 2006, MusicIP 2006
[10] W3C website, <http://www.w3c.org>
[11] Several authors, "Open Fingerprint Implementation Whitepaper", March 12th 2006, Music IP 2006
[12] MusicBrainz WIKI website, <http://wiki.musicbrainz.org/XMLWebService>
[13] MusicURI Project website, <http://semedia.deit.univpm.it/musicuri>
[14] José M. Martinéz, "MPEG-7 Overview", October 2004, ISO/IEC 2004
[15] eTantrum website, <http://www.freetantrum.org>
[16] Free Software Foundation website, <http://www.fsf.org/campaigns/drm.html>
[17] Microsoft Corporation website, <http://www.microsoft.com/security/glossary.mspx>

AMIEDoT: An annotation model for document tracking and recommendation service

ROBERT Charles Abiodun
LORIA- Campus Scientifique, B. P. 239, 54506
Vandoeuvre-Lès-Nancy, France
Tel : +33383593000
Email : abiodun-charles.robert@loria.fr

ABSTRACT

The primary objective of document annotation in whatever form, manual or electronic is to allow those who may not have control to original document to provide personal view on information source. Beyond providing personal assessment to original information sources, we are looking at a situation where annotation made can be used as additional source of information for document tracking and recommendation service. Most of the annotation tools existing today were conceived for their independent use with no reference to the creator of the annotation. We propose AMIEDoT (Annotation Model for Information Exchange and Document Tracking) an annotation model that can assist in document tracking and recommendation service. The model is based on three parameters in the acts of annotation. We believe that introducing document parameters, time and the parameters of the creator of annotation into an annotation process can be a dependable source to know, who used a document, when a document was used and for what a document was used for. Beyond document tracking, our model can be used in not only for selective dissemination of information but for recommendation services. AMIEDoT can also be used for information sharing and information reuse.

Categories and subject descriptors

Information systems design, Document tracking, routing, recommenders, Information retrieval, filtering, and extraction

Keywords

Information research, model, document, user, time, annotation

General terms

Management, design, documentation

1. Introduction

Annotation has been a very useful tool in transmitting ideas from man to man. Not only that an annotation convey the thoughts of an initial user of a document to another user of the same document, it testify to the use of that document. The importance of annotation as a tool in information management can be seen with its popularity. Many text processors like Microsoft word, Adobe Acrobat and the like integrate features that enable users to annotate electronic documents. We believe that electronic annotation made by different users should not only be restricted to interpretation of the content of document(s); annotation tools can be designed to assist in recommendation service, information management and document tracking. It was on this basis that an annotation model AMIE was conceived.

2. Background

Annotation can be perceived from different perspectives and can assume different forms but for our study, we will define it as an action and an entity. From the perspective of an action, annotation can be defined as an act of interpreting a document. The interpretation is of a specific context and is expressed on the document. The interpretation can be made by the producer of the document or another person. It should be noted that when a document author makes annotations on his own document, he is seen at that moment as a reader of that document and not an author. Considering annotation as an entity, we define it as written, oral or graphic information usually attached to a host document meant to attest to the use of a document, for evaluation or interpretation of a document. Our study here we dwell on these two definitions of annotation interchangeably.

Electronic annotation can not take place until after the document has been made available to its audience. Every annotation on incomplete document is considered as part of the initial document. This is important as we apply annotation to published work. Annotations will normally take a different form and different look as compared to the original document. The difference in look may be noticeable in form of character used, font, style, color or additional signs and images that is not characteristic of the original document. The common intercept between annotation and the original document is the medium of transmission.

3. Constituents of an annotation

An annotation is essentially consisting of three main components; the annotator (person making the annotation), the document being annotated and the resulting annotation itself. We will not give attention to the annotator in this study because our concern here is not on user modeling or profiling. A document is defined in a general form as a trace of human activities [15]. A trace of human activities can include archaeological artifacts, buildings, cinema, books and monuments. In another word, an archaeological artifact is a

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 145–150.

document as much as a building. Though our finding in most of the cases is applicable to documents of various types, attention is given to written documents.

A document essentially contain information meant for interpretation (read, viewed, heard, perceived) by a certain group of people. The audience may or may not be pre-determined. It is therefore imperative that a document be made available to its potential audience. A document itself may be in oral, graphics or text form. It may be tangible or intangible.

Annotation can not take place until after the document has been completed. An annotation is not a property of a document. For instance, a plate number of a vehicle is not an annotation though it is attached to a vehicle. This is because, we consider a plate number as a property of a vehicle. A vehicle is a complete entity with a plate number. Annotations will normally take a different form and look with respect to the original document. The different in look may be noticeable in form of character used, font, style, color or additional signs and images that do not form part of the original document.

From our study of the literature on annotation, we were able to identify the following reasons why annotations are performed:

- Add an explanation to a document section (definitions, examples, references, etc.)
- Provides a means of evaluating a document (relevance of a document by providing a global point of view or a detailed evaluation criteria)
- Associate specific interpretation to a section of a document or to the document in its entirety, by giving additional attribute to the document with an associated value
- It could be used as a medium of information sharing,
- It may serve as a means of sieving information.
- Means of interpretation of document,
- It is a means of creating a forum for independent view of document,
- Facilitate critical reasoning,
- Permit the user to construct a personal representation of the document,
- They can attest to a witness of personal commitment by a reader to a document ,
- Permit monitoring trace of document use,

It should be noted that annotation does not result in the modification of the initial document. It may however constitute a new document for the reader. This point is essential in the sense that the author's copyright is protected.

According to Bringay et al [5]. annotation helps in the legibility of information. Annotation may at one time make the document legible but may also hinder the legibility of the same document at another time. It does not necessarily aid in making the information clear but gives a specific interpretation to the information contained therein.

Annotations are performed by users who have the intention of storing their point of view for future reuse. Among the users (or readers) are students, researchers, lecturers, or the general public. Annotations can be made manually. For example, stickers or post-it can be scotched at specific pages of a book. Specific colours may be used to underline a section of a document in order to specify the importance of that section. It could also be in form of underlining. Text grouping with the use of brackets or braces is sometimes used to annotate. It may also be in form of passage or paragraph numbering.

With electronic software, it is possible to create manual annotations and also store them for future and more elaborate use.

A person making an annotation has an objective in mind. He is making annotation to achieve among others reasons: He could be describing (summarizing) or evaluating (analyzing) an informative resources based on standard criteria.

4. Existing models

The basic objective of annotation conception is to provide for additional set of information that was not specified by the initial author of the document. This information is saved to the original document and referenced by a link. The goal of annotation is to allow addition to existing resources by individuals who normally will not have direct control on the original document.

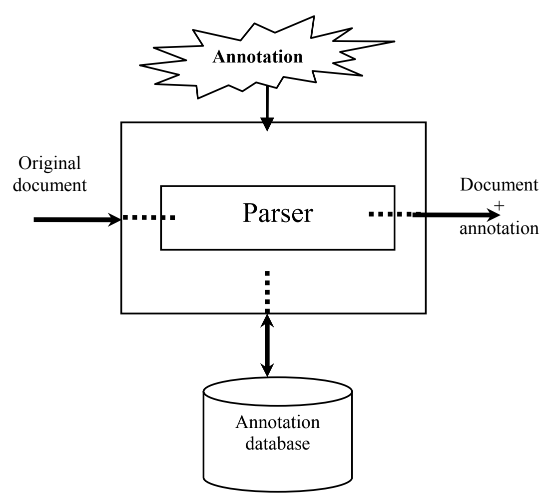


Figure 1: Architecture of generalized annotation system

Table 1: Document and user tracking based on log on documents (annotations)

	Document and user tracking	Fixed parameters			Representation	Value range [u=user,d=doc,t=time]
		User	Doc	Time		
1	Annotations of all users of all document				$\iiint dUdDdT$	$[0\leq u<\infty],[0\leq d<\infty],[0\leq t<\infty]$
2	Annotations of all users of all documents at a specific time			X	$T\iint dUdD$	$[t=1],[0\leq u<\infty],[0\leq d<\infty]$
3	Annotations of all users of a document		X		$D\iint dUdT$	$[d=1][0\leq u<\infty],[0\leq t<\infty]$
4	Annotations of all users of a document at a time specific		X	X	$DT\int dU$	$[d=1][t=1],[0\leq u<\infty]$
5	Annotations on all documents used by a user all time	X			$U\iint dDdT$	$[u=1][0\leq u<\infty],[0\leq d<\infty]$
6	Annotations on all documents by a user at a specific time	X		X	$UT\int dD$	$[u=1][t=1]\ [0\leq d<\infty]$
7	Annotations by a user on a document	X	X		$UD\int dT$	$[u=1][d=1][0\leq t<\infty]$
8	Annotation on a document by a user at a specific time	X	X	X	UDT	[u=1],[d=1],[t=1]

We can explain most models of annotation with figure 1. A document is sent to a parser with an annotation originating from the user of the system. The parser is considered as the motor of the system. How the annotation is made and to what part of document the annotation is addressed is what makes the difference. Generally annotation is added to the document based on a specific model. An annotation with the original document is created and returned to the user of the system. This created annotation is generally in form of the original document with a link (visible or invisible) pointing to the location of resulting annotation stored in an annotation database. The location of associated annotation database is also based on several factors depending on the level of security consideration. Some annotations are stored on the application server, which demand high security considerations. This type of system enhances optimal sharing of information between users but limits privacy. Some other annotation databases are stored on local machine. In this case, it enhances higher security but limit sharing of resources. A compromise between these two types of systems is the use of proxy-storage server. In this case another machine is situated in between the application server and the local machine to store annotations.

Several annotations systems were developed along the line of the structure of the document or based on the organization of the resulting annotation [1].[11]. Prominent among annotation system aimed at the structure of the document are annotation of type structuring and annotation for type classification. We can also consider annotations based on the methodology used in creating the annotations. Some are automatic, some semi-automatic and others are manual.

Annotation systems based on the structure of the document are concerned with the structural relationship between resulting annotation and the elements of the document annotated. Several annotation systems on the internet were conceived along this line. Example includes “annotation engine”, Hylight, AMAYA, YAWAS [9]. and CritLink [20]. Annotation tools such as the one in Microsoft word is of the type structuring. Some annotation tools were conceived to classify documents [6].[11]. The inside structure of documents are not addressed but the general concept or interpretation of the entire document as a whole. An example of annotation for classification is Furl (http://www.furl.net).

Annotations tools based on the methodology of creation generally give rise to semantic annotation, ontological annotation or linguistic type of annotation. Some annotations tools were considered as functional [13]. They can still be seen as either based on the organization of the resulting annotation or based on the structure of the document.

5. Our approach

Our approach is from the perspective that annotation attest to the use of a document. The annotation may not necessarily be placed on the document but kept in a separate database and a link provided between annotation database and the bibliographic references. The objective of providing the link is so that a base for document tracking and document management may be created. Specifically, we want to be able to analyze the feedback made inform of annotation. These feedback will provide information like when was a document used? Who used a particular document? For what was a particular document used?

Among likely-hoods, we observed that (a) dissimilar document can be used differently by one or more users. (b) two or more individuals will not make use of the same document the same way (c) the same user may not use the same document the same way at different time. A document can be used several times by a particular individual. We presume that one document can not be used by two users at a time. One user may use a

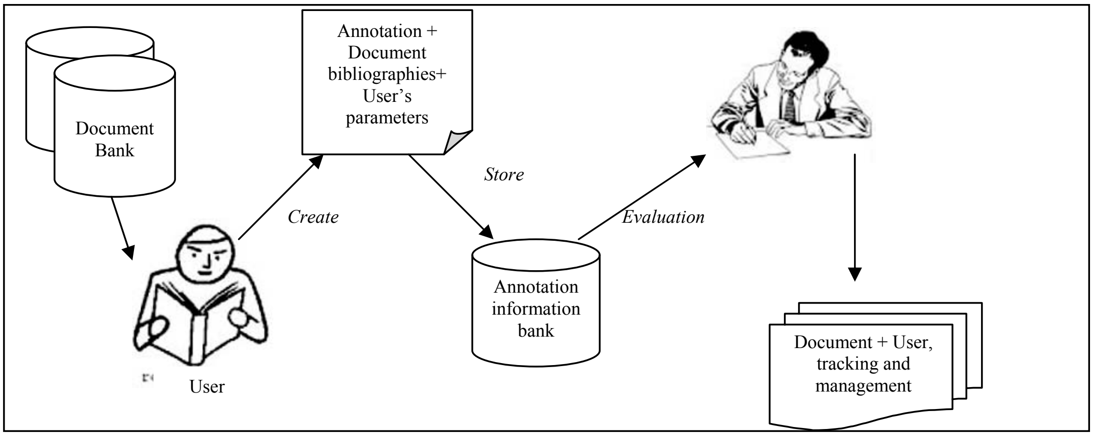


Figure 2: Application of AMIEDoT model for document and user tracking

document different compared to another user. A particular user may use the same document differently given a time frame. In applying annotation tool into document tracking and recommendation service, we are considering document use (annotation) in terms of users, documents and time. Series of (document use) annotations over time on one or more documents, by one or more users can be used to evaluate the use of document and interest of individuals.

An annotation or a set of annotation can be represented as

$$\iiint_x dUdTdD$$

Where dU is the variation in users, dT and dD are variations in time and documents parameters respectively. Specifically, we are signifying that annotation can be seen as a function of user (U), time (T) and document (D).

One or more of these parameters can be kept constant while the other varied. The three parameters when kept constant refer to a single case of an annotation. In the case where all these vary, it imply every possible annotation on a set of documents of interest.

We can be interested in the document use made by a particular user on a particular document over time. The objective of this may be to see his reaction or the user's disposition to an event. We can represent this as

$$UD\int dT$$

We can represent this in a three dimensional graph with each of the parameters in X, Y and Z axis respectively or with a table as in Table 1. We used the word annotation in this table to include the log of document used. This is because some document may not necessarily be annotated.

6. Specifications in AMIEDoT model

Our model consists of four main entities considered as the core of the model: (a) the user who is also the annotator, (b) the document in question, (c) annotation transaction and (d) the process of annotation creation. We address the model from these four perspectives. Each of these parts has its characteristics and properties. We attempt to describe each.

a. The user is the annotator

The user information is generally available when a user signifies his intention in the use of the library. They are normally stored differently from library databases.

The user is identified with the following parameters.

- Annotator's reference (*this is a unique reference that is used to identify a user*).
- identity (*identity of the document user*)
  - His name (*first name and last name*)
  - Email address
  - postal address (*or sectional address*)
  - region
  - age-group
  - country
  - social class
  - area of activity (*teaching, research, student etc*)
- session (*session is used to identify user's activities in the process with date and time*)

b. Document

The document consulted is paramount in any annotation process

- document title (*original title of document*)
- descriptors and keywords (*descriptors are words used to describe the document*)
- authors (*are the producers of the document, their names and surnames*)

- date of publication of document
- format of document (PDF, word, html etc)
- abstract / résumé

c. Annotation transaction (context of annotation stored in a storage)

This is meant to store the session of user every time the system is consulted.

- approach {the type of annotation i.e. follow up or new annotation}
- context reference
- session reference (date/time)
- implicit parameters of the user
    - Length on system
    - Documents consulted
- why was the document consulted?
    - Leisure consultation
    - Knowledge acquisition
    - Accidental consultation
    - Academic reading
    - Research reference
    - To answer a question
    - Historic reference
    - Internet link
    - Other reasons
- explicit parameters of the users
    - user name

d. Annotation creation

- reference (*is the reference, or code for future reference*)
- type (*the type of annotation used*)
    - » marking,
    - » typographic
        - ▪ italics , underlining ...
    - » reformatting of text using brackets and braces,
    - » passage numbering,
    - » text
        - ▪ in margin, footnotes, endnotes, in the gutter, by icons),
    - » icons
        - ▪ stars, question marks, exclamation marks,…
    - » symbols
        - ▪ to describe associations, relations between words..
- annotation location
    - ▪ left margin, right margin, footer, header, gutter, outside document, end of document
- why annotating (objective of annotation)?
    - » recapitulation, evaluation,
    - » summary, raise a point,
    - » classification, structuring,
    - » differentiating, for information,
    - » answer to a question,
    - » illustration, extension of document,
    - » clarify ambiguity of document

Application of the model

When a user request for a document from a document bank like in a library, usually, no record is made on the use of the document requested. We care considering the case where every user of every document in a library is recorded in term of the use of document, the period when document was used and annotations on documents used. This can be achieved by use of filling a page questionnaire. The form can contain very few questions like (a) Why the document was consulted? (b) Why was annotation made? and (c) and free form comment of users (annotation). Other parameters of the model can be provided by a librarian. In some of the cases, users may not provide any particular comment. Our interest is not necessarily on the comment he made, but on his profile and the consultation on the document and for what the document is used for. Because, most library users may not be willing to make comments, or assessment on every document used, for a start, for a start, we can apply this model to borrowed documents. We are aware that there exist necessary bibliographic information on all documents in libraries and database of users. It is the merging of bibliographic records with user's identity in time with additional comments that will provide vital information for document and user tracking.

A part from the fact that information on user and on document are the core of our model, we can find out among others the following information using the model: (a) The most consulted document (b) The most frequent objective of document consultation (c) the frequency of users in the library (d) Which user is in what domain? (e) Which user is related to another user from the perspective of the documents they consult? (f) What is the view of a user (or a group of user) with respect to a discipline - annotation of a user (group of users) With respect to documents in a discipline? (g) What social class consults most frequently? (h) What type of document is good for what user?

From the annotation made on document consulted, some of the information that can de derived include (a) The general interest of a user (or group of users to) (b) Type of document most interesting to a user (or a group of user)? (c) The most important objectives for making annotations? (d) The trend or general perception of a document or group of documents? Some of these analysis may be useful in classification or reclassification of document. It can even be used in associating key words and descriptions to document.

7. Perspective

The problem that has not been fully considered includes, how do we reconcile the changes that may exist in document with time in respect with usage? We are concerned with changes initiated by the author of the initial document. Of course, we can assume a different status for the "new document". The work on reference-based version model [7]. offer a great potential. It may be very interesting to see not just the use of document with time but the variation in document as well. A user may make use of a document differently if there are variations in the source document.

Our analysis does not consider intra-parameter considerations. For example, we may be interested in seen the effect of social class of users with type of annotation or document descriptions. These and other possible analysis is left to the discretion of the users.

8. Conclusion

From our studies, we have shown that annotation can be very useful in document tracking and user analysis. Annotation has been viewed as a function of its maker (the annotator), the document been annotated and the time of annotation. These three parameters are very important in its application to document/user tracking and management.

REFERENCES

[1] D. Bargeron, A. Gupta and A.J. Brush Bernheim., A Common Annotation Framework**,** Technical Report MSR-TR-2001-108, Microsoft Research, Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, 2001

[2] A.J. Bernheim Brush, Annotating Digital Documents for Asynchronous Collaboration, Department of Computer Science and Engineering, University of Washington, Technical Report 02-09-02, September 2002, 2002

[3] D. Bueno and David A., METIORE: A Personalized Information Retrieval System Source Proceedings of the *8th International Conference on User Modeling 2001*, in Lecture Notes In Computer Science; Vol. 2109, Pages: 168 - 177 , 2001

[4] N. Bouaka and David A., Modèle pour l'Explicitation d'un Problème Décisionnel: Un outil d'aide à la décision dans un contexte d'intelligence économique. *in Conférence "Intelligence Economique : Recherches et Applications"*, Nancy : 14-15 avril 2003. , 2003

[5] S. Bringay, C. Barry and J. Charlet, Les documents et les annotations du dossier patient hospitalier, « *Information-Interaction-Intelligence* » Volume 4, n°1, Juillet 2004, 2004

[6] R. Casallas-Gutierrez, Objets historiques et annotations pour les environnements logiciels, Thèse de Doctorat à l'Université Joseph Fourier Grenoble I, May 1996. , 1996

[7] S.Y. Chien, V. Tsotras and J. Zaniolo, Carlo Efficient Management of Multiversion Documents by Object Referencing, *Proceedings of the 27th VLDB Conference*, Roma, Italy, 2001

[8] A. David, D. Bueno, and P. Kislin, Case-Based Reasoning, User model and IRS. In *The 5th World Multi-Conference on Systemics, Cybernetics and Informatics* - SCI'2001. International Institute of Informatics and Systemics (IIIS). (Orlando, USA) , 2001

[9] L. Denoue, L. Vignollet, An annotation tool for Web browsers and its applications to information retrieval. *Proceedings of RIAO 2000* , 2000

[10] S. Goria and P. Geffroy, Le modèle MIRABEL : un guide pour aider à questionner les Problématiques de Recherche d'Informations.*In Veille Stratégique Scientifique et Technologique - VSST'2004 (Toulouse).* 2004. 5p. , 2004

[11] R. M. Heck, S. M. Luebke, and C.H. Obermark, A Survey of Web Annotation Systems, Internal report, Department of Mathematics and Computer Science, Grinnell College, USA, 1999

[12] P. Kislin and A. David, De la caractérisation de l'espace-problème décisionnel à l'élaboration des éléments de solution en recherche d'information dans un contexte d'Intelligence Economique : le modèle WISP, *in Conférence "Intelligence Economique : Recherches et Applications"*, Nancy : 14-15 avril 2003, 2003.

[13] B.K. Liisa, M. W. Gary, B. FZ Lang B. and G. Burgern, AutoFACT : Automatic Functional Annotation and Classification Tool, BMC Bioinformatic, 6:151, 2005

[14] H. Martre, « Intelligence économique et stratégie des entreprises », *Rapport du commissariat Général au Plan*, Paris, La documentation Française, pp 17,18, 1994

[15] Y. Prie, « Modélisation de documents audiovisuels en Strates Interconnectées par les Annotations pour l'exploitation contextuelle » Thèse de Doctorat à l'Université Claude Bernard Lyon1, France, 1999.

[16] A.B.C. Robert, Représentation des activités du veilleur en contexte de l'intelligence économique, DEA en Sciences de l'information et de la Communication, Université Nancy 2, Université de Metz, Octobre 2003, 2003

[17] M. A. Schickler, M.S. Mazer and C. Brooks, Pan-Browser Support for Annotations and Other Meta-Information on the World Wide Web, *Fifth International World Wide Web Conference* May 6-10, 1996 , Paris, France, 1996

[18] S. Sidhom,"*Plate-forme d'analyse morpho-syntaxique pour l'indexation automatique et la recherché d'information: de l'écrit vers la gestion des connaissances.*", Thèse de Doctorat à l'Université Claude Bernard Lyon1, France, Mars 2002. **,** 2002

[19] D.V. Sreenath, W. Grosky and F. Andres, *Intelligent Virtual Worlds: Technologies and Applications in Distributed Virtual Environments*, chapter Metadata-Mediated Browsing and Retrieval in a Cultural Heritage Image Collection. World Scientific Publishing Company, Singapore, 2002

[20] K.P. Yee, CritLink: Advanced Hyperlinks Enable Public Annotation on the Web. Demonstration abstract. ACM Conference on Computer-Supported Co-operative Work, , 2002

Intelligent Assistance for a Task-oriented Requirements Management

D. Mueller, P. Dietz, N. Mueller
Institute for Mechanical Engineering, Clausthal University of Technology
Robert-Koch-Str. 32
D-38678 Clausthal-Zellerfeld, Germany

***Abstract*-Requirement specifications for complex products are hard to handle due to their high amount of number and interrelations. Thereby, requirements represent the legally binding basis for the product development. In this paper we present a concept for a task-oriented filtering and provision of requirement specifications to support the engineer in his information management. The approach is based on a semantically enhanced categorization of requirements to prepare a computer interpretable information basis. The selection of processed requirements and their monitoring with respect to changes will be implemented by means of information agents. The associated agent schema and exemplary system architecture is provided. The approach enables an optimized processing of requirements which backs an efficient product development.**

I. INTRODUCTION

It is essential for the manufacturing industry to bring new products in short time at low costs and of high quality onto the market. This implies that a company has to act efficiently and flexibly if it wants to survive in the global market. Innovations represent the unique selling point in competition [1]. Innovative solutions are primarily realized by the extended application of electronics and software in automotive or aerospace industry. This is associated with an increased product complexity, which is characterized by the type, diversity and number of elements and relations, as well as the dynamics of the system [2]. The development of innovative products is linked with the integration of different processes and domains. The control, management and implementation of such an integrated product development is one of the challenges that have to be tackled nowadays.

It is commonly known, that the product development is primarily responsible for the determination of a product's total costs. Thereby, requirements represent the legally binding basis for the development tasks. A requirement is an expression of a perceived need that something be accomplished or realized [3]. This definition by Gabb et al. includes the demands and wishes a desired product has to fulfill, as well as the constraints regarding e.g. system environment, services or personnel entities.

Three abstract user profiles can be identified that are concerned with requirements. The *developer* is the recipient of a requirement. He is directly responsible for the problem solving and component design with reference to the stated requirements. The *system analyst* is responsible for the general requirements engineering and system design in the early product development phases up to the start of production. He is engaged in the extensive elicitation, analysis, negotiation and documentation of qualitative requirements, as well as the fundamental system conception. The *stakeholder* characterizes a person with a not explicitly defined involvement along the product development. This can be a supplier participating in the requirements elicitation and negotiation process, or a person in charge with access to the requirements specification to support his tasks (e.g. marketing).

The design process can be characterized as follows. It is definitely personal, based on creativity, and dynamic. The agreement on requirements and problem-solving solutions is marked by negotiations and compromises. The engineers' tasks are based on their individual knowledge by interpretation of available and acquired information. Several systematic approaches to engineering design have been proposed (among others [4], [5], [6], [7]). Despite the variances, general tasks are common to all these approaches:

- Requirement specification and planning
- Search and development of solutions
- Selection and optimization of variants

Requirements have a relevance in all these tasks. After determination, their fulfillment and adaptation have to be considered continuously [8]. The procurement of information changes its focus in the course of the development cycle [9]. It is mainly problem-oriented in the beginning (What has to be created?), and alters to solution-oriented to the end (Does my solution fulfill all demands?). The aforementioned user profiles are also characterized by different informational needs. Additionally, the view on information is influenced by the respective task, e.g. regarding the level of detail.

The statement of requirements represents a problem regarding their qualitative documentation, especially in matters of clearness and analysis. Nowadays, the specification of requirements results still predominantly in a natural language based format. Model-based or graphic approaches are not widely distributed yet in the manufacturing domain. The application of text is linked with problems of e.g. incompleteness, inconsistency or ambiguity [10]. This restricts also a computer-based processing. Additionally, the increasing product complexity affects the number of requirements and the level of interrelations. This has an aggravating influence on their processability.

The identification of relevant requirements has a positive influence on the quality of the generated results [11]. Nevertheless, the complexity of the requirements document raises the need for a specific support, as a manual analysis and processing is extremely time-consuming respectively hard to real-

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 151–156.

ize. This is supported by the psychological point of view. The human ability to handle a great amount of information is limited, which leads to an incomplete or reduced consideration [12]. Requirements of high quality and the opportunity of their goal-oriented processing will meet the problem of information overload and support the necessary systematic proceedings [13].

The approach to enable a supported provision of only those requirements that are relevant for a specific task leads to the objective of a flexible, task-oriented requirements filtering. The user should be able to extract requirements from a database-driven requirements document by specification of a task-specific retrieval request. Additionally, the user should be notified on relevant changes in requirement specifications to always act on the latest status of requirements. This problem entails:

- Intelligent analysis and selection of requirements,
- Task-oriented editing and processing of requirements,
- Continuous, flexible requirements management,
- Change monitoring.

The remainder of this paper will describe the conceptual approach for the task-oriented requirements management by intelligent assistance regarding requirements filtering in detail, followed by the specification of the associated agent system, and conclude with the synopsis and evaluation of the proposed concept.

II. Concept Development

The functions that will be fostered by the introduced approach focus on the coverage and control of the complexity in requirements specifications, enable an efficient search and selection of requirements, and can be flexibly and continuously adopted by users during their variable tasks. Fundamentals to this method are a semantically enhanced categorization of requirements, information model integration, and the adoption of information agents for the task-oriented analysis, filtering, and monitoring of requirements.

With reference to Belkin and Croft, the problem of information filtering can be identified as the selection of information relevant for an individual user [14]. The qualification of information agents can be gathered from their definition. Klusch characterizes an information agent as an autonomous, computational software entity that has access to one or multiple, heterogeneous and geographically distributed information sources, and which pro-actively acquires, mediates, and maintains relevant information on behalf of users or other agents preferably just-in-time [15].

The basic skills of an information agent are divided in communication, collaboration, knowledge, and low-level tasks [16]. Communication can be accomplished with information systems including databases, agents or users. The interaction respectively collaboration with users or agents can be established on higher-level. Data, information or knowledge of different formats can be processed by the information agent, including ontological knowledge, metadata, profiles and natural language. Main tasks regarding the handling of information involve its retrieval, filtering, integration and visualization. The basic skills are interrelated and form the specific capability of an information agent.

The general applicability of information agents still asks for a task-specific configuration and determination regarding agent, environment, and information basis. As mentioned before, the processing of natural language by IT-systems is limited. This includes especially the identification of relevant information. It is necessary to cope with the vagueness and complexity of semantic relations of requirements. Therefore, an enhanced organization of requirements is necessary based on a formal specification mechanism, which can be achieved by a categorization of requirements, in accordance with Gabb [3].

It is mandatory that the semantic relationships will be covered by a defined schema. By mapping of requirements contexts, the efficiency regarding their IT-supported analysis will be improved significantly. The semantic allocation of data is of high complexity and within the focus of several actual research initiatives, e.g. the development of the Semantic Web[1] is of high importance in this area. Thereby, the integration of ontologies as formal specification mechanism has been considered. Gruber specified a commonly agreed definition, whereas an ontology is an explicit specification of a conceptualization [17]. An ontology extends the linguistic means of expression of a corresponding representation [18]. A basic product ontology refers mainly to classification systems [19]. Methods that are used by ontology integration approaches are e.g. text similarity, keyword extraction, structural analysis, and data interpretation and analysis [19].

The generation of a classification system for requirements solely realized per ontology requires immense efforts due to the existing complexity and diversity. A reduced system would limit the functionality and flexibility. Due to this fact, the integration to the standardized systems engineering reference model ISO 10303 STEP AP233 has been taken into account, pre-published as publicly available specification ISO/DRAFT-PAS 20542:2004(E) [20]. The following is within the scope of the AP233 model:

- Products with conformity to the concept of a system,
- System definition data and configuration control data pertaining to the design and the validation phases of a system's development,
- Requirements,
- Functional analysis data including functional behavior specifications,
- Physical architecture and synthesis data providing a high level view on the system under specification,
- Elements that are used to represent and trace requirements and functional allocation.

Fig. 1 shows a conceptual view of the AP233 system model in UML syntax, whereas every box represents a group of related entities. The central unit of functionality for requirements allows the integration of a classification system, besides the general representation of requirements, their interrelations and assignment to system specifications. The standard

[1] see http://www.w3.org/2001/sw/

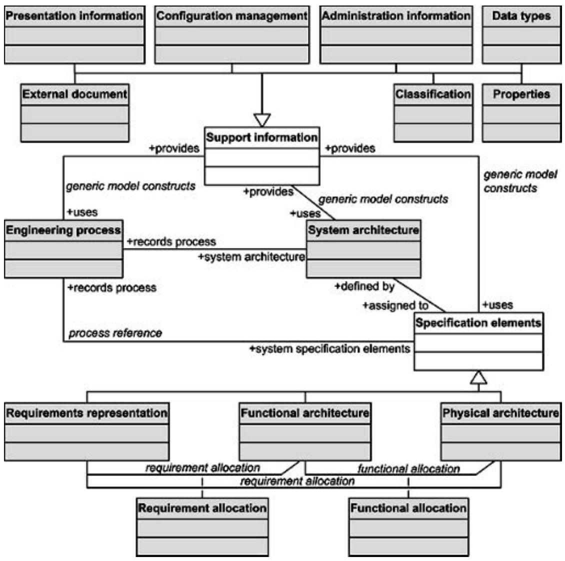


Fig. 1. Conceptual view of AP233 as described in [21]

does not specify a fixed structure for requirement classification. A dynamical structure is provided instead by requirement classes and relationships. The semantics of requirement relationships will be covered by determination of explicit characteristics.

The proposed extension of the AP233 reference model offers a great potential, which has not been exploited before. The mature, comprehensive requirements categorization adds a fundamental, semantic component to the information model. Additionally, the direct link of classification schema, requirements specification, and furthermore other system components enables an extended, flexible, complementary arrangement including an improved processability by IT systems.

Set up on a structured, interpretable information basis, intelligent algorithms can be applied to deliver better, more reliable results regarding their advanced, selective tasks. The representation of categorized requirements promotes the mapping of existing relations, derivations or subdivisions. This fulfills the premises for an assisted detailing of requirements. The user will be able to make an goal-oriented inquiry about requirements.

The enhanced, categorized requirements respectively system specification model represents the operative basis for a task-oriented requirements management. Nevertheless, it is necessary to prepare the present requirements accordingly. The defined process of a classifying formalization is responsible for the transformation and categorized organization of natural language based requirements. Fig. 2 displays the relevant steps.

The lexical classification (node A1) includes the content related analysis of the natural language requirements regarding explicit characteristics. The resulting requirements model, initially extended by content-based classifications, will be completed within the following semantic classification (node A2). This step is responsible for the identification of implicit requirements characteristics and interrelations. The accordingly produced requirements model fulfills all qualifications for a user initiated, intelligent assisted requirements analysis and selection (node A4). The entire process delivers filtered requirements as result, subject to a specified inquiry. The output will be presented to the user in the form of partial requirements lists.

The process of the classifying formalization can be optionally extended by application of an enhanced requirements formalization (node A3) according to Heimannsfeld [10]. This formalization is responsible for the transformation of natural language based requirements to a model based representation by systematic identification of linguistic and grammatical elements. The extensive generation of a formally enhanced requirements model promises improvements regarding the outcome of analyses and more precise information on details. Such specific type of information is not primarily in the focus of the aforementioned user profiles. Thus, it has to be balanced if the extra effort is reasonable.

Nevertheless, the standardized, model-based approach of the concept allows a variable, scalable implementation depending on respective needs. It should be noted, that the tasks of nodes A1 to A3 are carried out by human interaction, usually by the system analyst. These tasks require in-depth knowledge that is so far not reproducible or deducible by current IT systems. The system of information agents is responsible for the filtering of accordingly organized requirements. The design of an exemplary agent system for the realization of the proposed approach will be described in detail in the following section.

III. AGENT SYSTEM

The accomplishment of complex tasks regarding filtering, monitoring, and management of complex, numerous requirement specifications comprises the core functionality that has to be covered by the agent system. The systematic methodology according to [22] was chosen for the applied system de-

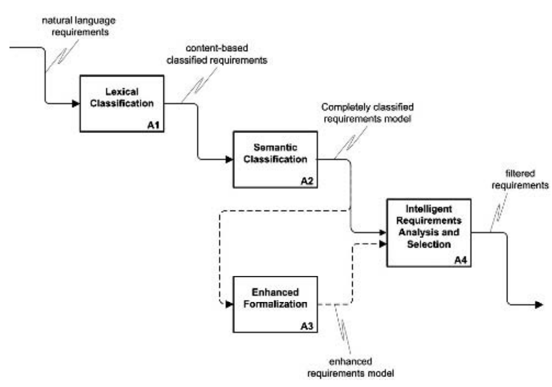


Fig. 2. Process of requirements formalization and analysis

velopment. This approach aims for the use of the JADE (Java Agent DEvelopment framework) platform in particular, including their inherent agent management concepts. JADE is a software framework by TILAB[2] (Telecom Italia Lab) incorporating Java technologies. The enclosed middle-ware complies with the FIPA[3] (Foundation for Intelligent Physical Agents) guidelines and supports the implementation and control of the agent system by provision of a basic agent architecture.

The modeling of the use case for the task-oriented requirements management and the systematic analysis of necessary agent tasks, interaction, and behavior based thereon resulted in the determination of four agent types. Thereby, security issues have not been considered, as well as agents for the general agent management non-specific to the problem. The overall concept of the system is displayed in Fig. 3.

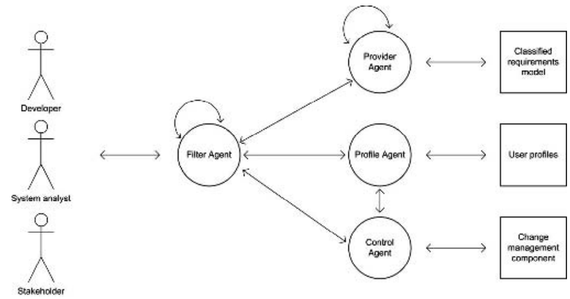


Fig. 3. Schema of the agent system

Besides the agent types indicated by the circle, the aforementioned user profiles indicated by the actor symbol and three external resources indicated by the rectangle belong to the system. Acquaintances of components that require an interaction during execution are represented by the double-headed arrow.

The users of the system initiate the task-specific requirements retrieval by specification of search attributes and consideration of user related preferences. The agent system works off autonomously the request. The database of the requirement model constitutes the central information basis. Additionally, information on the user profiles will be managed and maintained in a database. The change management component is responsible for the system monitoring.

The Filter Agent takes on the interface function among user and system components. It represents the central point of organization and coordination of the task-oriented requirements filtering. Based on the conveyed search criteria specified by a request and extended by general user preferences, relevant categories, attributes or model elements have to be identified and analyzed for the requirements selection. The determination results in the transfer of partial, goal-oriented search requests to a Provider Agent. The return values of the multiple queries have to be correlated regarding e.g. relevance, redundancy or consistency. The disclosure of ambiguous or incomplete results will cause renewed queries. The filtered requirements will be presented to the user with finalization of the query. A user related rating of the requirements regarding relevance optimizes further filtering processes. Query and result will be stored index-based in the user profile to enable traceability regarding updating.

The Provider Agent establishes the connection to the classified requirements model. The main task is to answer the queries. The Provider Agent directs the query to the database. Furthermore, it is able to perform task-oriented analyses of query and selected requirement attributes on his own and to activate specific sub-retrievals. This enables among others an additional tracing of structural requirements regarding insufficient classification, an extended examination of interface specifications, or an identification of relevant relations.

The Profile Agent represents the interface to the master data of the user profiles. This agent type coordinates the user related information management. This includes the allocation of the user profile at login, maintenance of user preferences and properties, and securing of indexed queries. In addition, requests from Filter Agent or Control Agent have to be served.

The Control Agent is responsible for the change monitoring of requirements. The completion of requirement adjustments will be registered based on a change management. Possible approaches are the setting of a change flag or the continuous surveillance. The notification process will be initiated autonomously. This incorporates the investigation of persons that have to be notified on a modification on the basis of stored query indexes or assigned person attributes. An identification causes a notification (e.g. news service, email) of the respective person, inclusive the transmission of the updated requirement. It will be possible to transfer directly the requirement to the task-oriented portfolio including an update of the linked query index, or to start another query.

The filtering of requirements is a complex, iterative process. Due to the variety of attributes that have to be investigated, it is not possible to realize it in only one, combined query, even by categorized requirements. This is another argument for the application of information agents. Filter Agent and Provider Agent are mainly responsible for the pro-active generation of partial queries by their agent types, which can be identified by the self-referencing arrows in Fig. 3. The system conception needs some more detailing, e.g. regarding definition of explicit behaviors or interaction patterns, but this should be out of the scope of this paper.

An exemplary architecture for the agent system will be presented below. The main components can be divided in the system for an IT-based requirements engineering and management, the agent platform and the user interface, as displayed in Fig. 4. The proposed architecture still features degrees of freedom regarding the implementation.

[2] http://jade.tilab.com/
[3] http://www.fipa.org/

The requirements management system (e.g. Doors XT[4]) is the central application for the documentation and organization of requirements. The specified information model has to be supported, e.g. per configured interface for the data exchange. The model-based classification and transformation of natural language based requirements can be realized by functional integration or by linking to an external system, that is the preferred solution. Such a system has been developed within the European research project KARE[5] and can be integrated by simple extension regarding the categorization. The tool *demanda II* is based on an ECCO C++ standard library as application instance, generated by the ECCO Toolkit[6] of PDTec GmbH. Additionally, the modules of change and user profile management have to be integrated. The provided APIs can be used for the access by the information agents.

The agent system performs the analysis and filtering of requirements as well as their monitoring. The decision for the Jade platform has been justified before. The initialization of the platform incorporates the immediate generation and establishment of the default administrative components. The Jade platform is not restricted to only one host, as can be gathered from Fig. 4. The distributed execution is based on an application instance in form of the Java Virtual Machine (JVM). Each JVM corresponds to one agent container per host with a functional run-time environment for a parallel implementation. The task-specific customization of the agent environment is only necessary for the communication and behavioral patterns of the application agents, as well as the interface functionality. Every container based on a Java run-time environment (JRE) features the defined application agents with underlying class schema and AP233 semantics.

The general utilization of the system in terms of selected requirements retrieval and change information service will be handled by the user component of the system. This constitutes the main interface visible for the common user, which is responsible for the behavior controlling interaction of agent system and user. Thereby, it is possible to implement the graphical user interface (GUI) as local application on a workstation or web-based via a web client using a standard internet browser. The realization is supported by appropriate programming languages, such as Tcl/Tk or Java. It is not necessary to force an integration to the requirements management system, as the overlap within the user groups has to be assumed small. This should be considered only with respect to further developments of distributed requirements management solutions. The main focus of the development for this component is on the interface specification. The exemplary implementation uses JavaServer Pages technologies for the web-based solution respectively Tcl/Tk or the Abstract Window Toolkit[7] (AWT) for the machine-based approach. Nevertheless, the web-based approach will be favored with respect

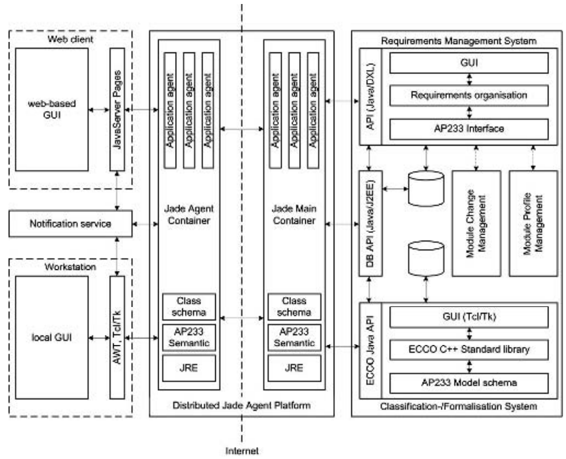


Fig. 4. Conceptual architecture

to best possible extensibility and connectivity to further agent systems. Additionally, possible mobile solutions will be supported. The notification service is a relevant feature of the user component. Possible solutions are email or news service, as mentioned before.

IV. CONCLUSION

The conception of a system for a task-oriented filtering and provision of complex requirement specifications has been introduced in this paper. The realization of the approach will be achieved by comprehensive categorization of requirements, incorporating important semantic aspects, to provide a computer-interpretable information basis. On this groundwork, an agent-based filtering and monitoring of requirements is applicable to assist the user in his tasks by an easier, advanced and timesaving information management. The optimized, flexible and individually oriented processability of complex requirement structures improves their integration and consideration during the system respectively product development. The system concept supports also the general accessibility of requirement specifications by all persons involved in modern, distributed development environments. The application of the approach will lead to a mature, efficient and less defective product development.

The detailed investigation of the task-oriented requirements management is presented in [23]. The relevance of the concept has been verified with the help of three case studies from different industrial areas. The characteristic activities of a task-oriented requirement categorization, their intelligent filtering, as well as the value-added, practical integration in the development process have been demonstrated. It is possible to master the complexity of requirements by semantically enhanced, categorized structures and hence intelligent interpretation. The integration of the approach can be achieved without major efforts, as just the requirements engineering process has to be adapted regarding the categorization. A domain-specific customization is recommended as well as the statement of guidelines, e.g. regarding scope and preciseness. The

[4] http://www.telelogic.com/index.cfm?
[5] KARE - Knowledge Acquisition and sharing for Requirement Engineering, EU ESPRIT No. 28916, 1998 - 2001, http://www2.imw.tu-clausthal.de/kare
[6] http://www.pdtec.de
[7] AWT in combination with Swing offers a class library for the Java related GUI programming

extent of the implemented categorization and the quality of the enquiry affect fundamentally the relevance of the results.

ACKNOWLEDGMENT

The Institute of Mechanical Engineering, Clausthal University of Technology, is partner of the EU-funded FP6 Innovative Production Machines and Systems (I*PROMS) Network of Excellence which is funded by the European Commission under the Sixth Framework Programme (FP6-500273-2). The authors would like to thank the Commission and the partners of the I*PROMS Network of Excellence for their motivation, contribution and support.

REFERENCES

[1] A. Radon, "Enabling innovation - integrating process," editorial, *Produktdaten Journal*, vol. 13, no. 1, *ProStep iViP e.V., Darmstadt*, July 2006, p. 3

[2] U. Lindemann, "Methodische Entwicklung technischer Produkte - Methoden flexibel und situationsgerecht anwenden," ISBN 3-540-14041-7, *Springer Verlag Berlin et al.*, 2005

[3] A. Gabb (Ed.), "Requirement categorization," *Requirements Working Group of the International Council on Systems Engineering*, *INCOSE*, 04 February 2001

[4] G. Pahl, W. Beitz, "Engineering design - a systematic approach," ISBN 3-540-19917-9, *Springer Verlag Berlin et al.*, 1996

[5] R. Zuest, "Einstieg ins Systems Engineering - systematisch denken, handeln und umsetzen," ISBN 3-85743-990-4, *Verlag Industrielle Organisation, Zurich*, 1997

[6] VDI[8] 2221, "Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte," *VDI Verlag, Dusseldorf*, 1993

[7] VDI 2206, "Entwicklungsmethodik für mechatronische Systeme," *VDI Verlag, Dusseldorf*, 2003

[8] N. Dylla, "Denk- und Handlungsablaeufe beim Konstruieren," ISBN 3-446-16327-1, *Hanser Verlag Munich*, 1990

[9] W. Daenzer, "Systems Engineering - Leitfaden zur methodischen Durchfuehrung umfangreicher Planungsvorhaben," ISBN 3-85743-833-9, *Verlag Industrielle Organisation, Zurich*, 2nd ed., 1978/79

[10] K. Heimannsfeld, "Modellbasierte Anforderungen in der Produkt- und Systementwicklung: Von Dokumenten zu Modellen," ISBN 3-8265-9277-8, *Shaker-Verlag, Aachen*, 2001

[11] J. Günther, "Individuelle Einflüsse auf den Konstruktionsprozess - Eine empirische Untersuchung unter besonderer Berücksichtigung von Konstrukteuren aus der Praxis," *Shaker Verlag, Aachen*, 1998

[12] G. Fricke, "Konstruieren als flexibler Problemloeseprozess - Empirische Untersuchungen ueber erfolgreiche Strategien und methodische Vorgehensweisen beim Konstruieren," *VDI-Verlag, Dusseldorf*, 1993

[13] P. Badke-Schaub, E. Frankenberger, "Management kritischer Situationen - Produktentwicklung erfolgreich gestalten," ISBN 3-540-43175-6, *Springer Verlag Berlin et al.,* 2004

[14] H. Sevay, C. Tsatsoulis, "Agent-based intelligent information dissemination in dynamically changing environments," in Intelligent Agents and their Applications, Jain, Chen, Ichalkaranje (eds.), ISBN 3-7908-1469-5, *Physica-Verlag Heidelberg et al.*, 2002, pp. 1-26

[15] M. Klusch, Information Agent Technology for the Internet: A Survey, Journal Data and Knowledge Engineering, special edition Intelligent Information Integration, Fensel (ed.), vol. 36(3), Elsevier Science, 2001

[16] M. Klusch, H.-J. Buerckert, P. Funk, A. Gerber, C. Russ, Applications of Information Agent Systems, *Intelligent agents and their applications*, Jain, Chen, Ichalkaranje (eds.), ISBN 3-7908-1469-5, *Physica-Verlag Heidelberg et al.,* 2002, pp. 217-248

[17] T. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," technical report KSL 93-04, *Knowledge Systems Laboratory*, *Stanford University*, 2003

[18] Y. Alan, "Integrative Modellierung kooperativer Informationssysteme - Ein Konzept auf der Basis von Ontologien und Petri-Netzen," *University Duisburg-Essen, Essen*, 2005

[19] S. Abels, L. Haak, A. Hahn, "Identifying ontology integration methods and their applicability in the context of product classification and knowledge integration tasks," report no. WI-OL-TR-01-2005, ISSN 1861-3748, *University of Oldenburg, Business Information Systems*, 2005

[20] International Standardization Organization, ISO TC184/SC4/WG3 N1355, ISO/DRAFT-PAS 20542:2004(E) Industrial automation systems and integration - reference model for systems engineering, 2004, http://www.tc184-sc4.org/sc4ndocs/n1740/

[21] E. Herzog, A. Toerne, "AP-233 architecture," proceedings *INCOSE 2000*, *10th annual international symposium of the International Council on Systems Engineering*, pp. 815-822

[22] M. Nikraz, G. Caire, P. Bahri, "A methodology for the analysis and design of multi-agent systems using JADE," http://jade.tilab.com/doc/JADE_methodology_website_version.pdf, 2006

[23] D. Mueller, "Intelligente Unterstuetzung für ein aufgabenorientiertes Anforderungsmanagement in der Integrierten Produktentwicklung," *PH.D. thesis, Clausthal University of Technology*, unpublished

[8]Verein Deutscher Ingenieure – Association of German Engineers

Translation of Safety-Critical Software Requirements Specification to Lustre

Dongchul Park
6-240 EE/CS Building
University of Minnesota
Minneapolis, MN 55455
park@cs.umn.edu

***Abstract* - SpecTRM-RL (Specification Tools and Requirements Methodology-Requirements Language) is a modeling language for describing safety-critical software requirements. However, SpecTRM-RL does not support formal verification, which plays a very important role in developing safety-critical systems and software. Lustre is a dataflow synchronous language designed for programming reactive systems. Lustre supports the analysis and formal verification as well as code generation. Therefore, by translating SpecTRM-RL into Lustre, it not only will endow verification function to SpecTRM-RL, but also will make it possible that SpecTRM-RL supports various analysis approaches of codes by using previously developed translator which converts Lustre into NuSMV, PVS, and SAL.**

In this paper, I present the rules to translate SpecTRM-RL to the Lustre language, and also present an empirical study in which we practically translate a SpecTRM-RL requirements document into Lustre using the rules proposed. This study shows that SpecTRM-RL can be effectively converted into Lustre so that it can support formal verification.

I. INTRODUCTION

Compared to a decade ago, software is widely used in safety-critical fields such as aircraft, cars, and medical instruments. So both finding errors early in the software development cycle with specification languages such as $RSML^{-e}$ and SpecTRM-RL and software verification with analysis tools such as PVS (Prototype Verification System) and NuSMV (a new Symbolic Model Verifier) play a pivotal position in the software development [1].

SpecTRM (Specification Tools and Requirements Methodology) is a toolset to assist the development of software-intensive safety-critical systems and software [2]. Thus, SpecTRM emphasizes system requirements and specification because most decisions that affect safety are made early in the product life cycle [3]. SpecTRM-RL (SpecTRM-Requirements Language) is a modeling language for describing such requirements. However, SpecTRM and SpecTRM-RL do not support formal verification which plays a very important role in developing safety-critical systems. This is their most critical weak point of them with respect to safety-critical system development.

Lustre is a dataflow synchronous language designed for programming reactive systems such as automatic control and monitoring systems [4][5]. Lustre supports the analysis and formal verification; so if we translate SpecTRM-RL models into Lustre codes, we can give SpecTRM-RL the features of verification and analysis Lustre originally has. Lustre is also widely used as an intermediate language in translations because we can easily and efficiently parse and manipulate the Lustre codes [6]. In particular, since the translator from Lustre to NuSMV, PVS, and SAL (Symbolic Analysis Laboratory) codes has been already developed by the Crisys group at the University of Minnesota [7], only if we translate SpecTRM-RL into Lustre, can the translated codes be automatically converted into other codes for various analysis tools such as NuSMV, PVS, and SAL [8] [9]. This not only endows a formal verification function to the SpecTRM-RL model but also makes it possible for the SpecTRM-RL model to support various analyses by using those analysis tools.

In this paper, I present the rules to translate SpecTRM-RL into Lustre, and also present an empirical study in which I translated two specific SpecTRM-RL models (Altitude Switch and Cruise Control) into Lustre with the rules I proposed. I also performed black box testing to verify both results. The studies show that SpecTRM-RL can be effectively converted into Lustre.

The main contribution of this paper is the development of a set of rules to translate a safety-critical software requirements specification (SpecTRM-RL) model to Lustre. As a result, this translation enables the SpecTRM-RL model to have formal verification as well as various kinds of analytical approaches by using the previously developed translator, which, ultimately, makes SpecTRM a more powerful toolset for safety-critical systems and software development.

This paper is organized as follows. In section 2, related studies are briefly discussed. Section 3 gives the translation rules I proposed and several examples with BNF grammar, and section 4 describes translation issues and approaches to address those issues. Section 5 explains the experimental setup and results, and finally, section 6 gives the conclusion on the problem.

II. RELATED WORK

A. SpecTRM-RL

As mentioned in the introduction, SpecTRM-RL is a modeling language used in SpecTRM to describe the behavior of system components. Although this language was initially developed to describe reactive embedded control system

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 157–162.

software, it has been successfully applied to other kinds of components including hardware [10]. SpecTRM-RL models are very readable and reviewable because the syntax of the language is regular and simple; so various kinds of stakeholders including domain experts, engineers, and managers can review this model with no specialized training.

A SpecTRM-RL model is a collection of model elements that describe the behavior of a component [11]. Even though model elements consist of these several elements, all of the model elements have some common traits; first of all, all begin with a declaration in a box at the top of the elements telling what kind of model element definition follows. Second, the next line shows the name of the model element. Third, the next part of the model element is a set of attribute-value pairs. Lastly, below the attribute-value pairs every model element has some kind of definition (See Fig.1).

Model elements

A SpecTRM-RL model consists of 6 model elements that describe the behavior of a component: outputs, modes, states, macros, functions, and inputs.

Each output model element describes the output behaviors of the system. So each output describes the value of the output as well as the conditions with which the output is triggered. In particular, output model elements begin with triggering conditions as an output definition which is an AND/OR table; so when the triggering conditions holds, the output is sent, otherwise, it is not sent.

All components, typically, will have major operating modes that strongly affect the way it responds to inputs. SpecTRM provides mode model elements to depict such groupings of behaviors.

SpecTRM-RL includes a model element for describing the inferred system state used by the controller to trigger outputs. The definitions of states, much like modes above, consist of several transitions. Every state, especially, must have a transition to 'unknown'. Lots of accidents tend to occur because the automated controllers do not have any requirements to handle unknown cases.

In order to abstract common logic and to increase readability and understandability, macros are generally used in the SpecTRM-RL model. Macros take a piece of an AND/OR table from another part of the model and endows it with a name; so the definition of macros consists of a single AND/OR table evaluating them as true or false.

If there are complex calculations that cannot be represented within an AND/OR table, it is hard to handle such calculations. To solve this problem, SpecTRM-RL provides a function definition language. The body of the function is composed of a series of statements that perform the calculation. Each function must have certain attributes which detail when it will be evaluated, what type of value it returns and what arguments it takes as well as what their types are.

In most systems, inputs come from sensors which measure and evaluate values from the controlled process or from the operator controls. Inputs have a type of integer, real, duration,

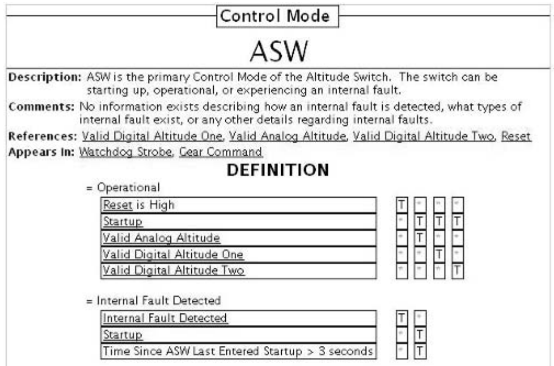


Fig. 1: A sample part of SpecTRM-RL model (mode element)

or enumerated sets. Inputs always have to transition to one of the following: new value, current value, or obsolete. In particular, the 'obsolete' transition is required of every input because many of the accidents thus far have resulted from software acted on data which is too old to be reliable.

B. Lustre

Lustre is a dataflow synchronous language designed for programming reactive systems. This supports their analysis and formal verification. In addition, Lustre is also widely used as an intermediate language in translations because it is parsed and manipulated easily and efficiently[4][12].

Streams and nodes

Lustre operates on streams constituted by a finite or infinite sequence of values and an associated clock defining the logical instants when the stream is active [13]. So the Lustre program can be seen as a set of equations operating on streams; any variable and expression denote flow; in other words, a pair which is composed of a possibly infinite sequence of values and a clock representing a sequence of times. A flow takes the n-th value of its sequence of values at the n-th time of its clock.

Lustre programs are composed of nodes which are self-contained modules with inputs, outputs, and internally declared variables. Lustre nodes are similar to functions or methods in other languages such as C and Java; however, they do not interact with global variables and values. Instead, they can retain their own state information because Lustre has a 'pre( )' expression which describes their previous value or state information. Thus, in Lustre, it is not necessarily guaranteed that the same inputs to a node will yield the same outputs. However, it always guarantees that the same input trace will yield the same output trace.

Time and Clocks

In particular, the concept of 'clock' is of consequence in Lustre. The Lustre language is basically based on the concept of logical time which is an infinite, well ordered sequence of logical instants [13]. So it should be noted that the concept of

Table 1: Lustre codes for new clock

```
node NEW_CLOCK (…) returns (…);
var
        X, Y, Z: int;
        A, B: bool;
let
        X = 0→ (pre(X) + 1);
        A = true→ (not pre(A));
        Y = X when A;
        B = (true when A)→ (not pre(B));
        Z = X when B;
tel
```

Table 2: Basic clock and newly defined clocks

X(Basic Clock)	0	1	2	3	4	5	6
A	True	False	True	False	True	False	True
Y	0		2		4		6
B	True		False		True		False
Z	0				4		

'clock' is not necessarily bounded to physical time; we can also either define whatever new slower clock we want or use the basic clock in Lustre. Because Lustre has Boolean-valued streams, the slower clock can be defined as the sequence of times at which the stream takes the value true. However, it is possible to create a notion of physical time by defining a Boolean input that represents some timescale (e.g. millisecond). The program responds to time inputs by toggling that input to match an external notion of physical time [14]. Therefore, the concept of the logical clock in Lustre lets the users abstract the real time and allows an easy and efficient development and verification of their applications [13]. Table 1 shows the Lustre codes and Table 2 clearly shows well basic clock as well as the new slower clocks defined by Table 1.

III. TRANSLATION RULES

This section defines the translation scheme for SpecTRM-RL constructs and each model element in detail with BNF grammar. The annotation in the above the table gives the BNF grammar for a piece of SpecTRM-RL specification, and the bottom table gives the equivalent Lustre translation for that example with BNF grammar. Note that due to space limitations, I did not provide a complete BNF by terminals.

Translation Scheme

Unlike other languages consisting of source codes for execution, the SpecTRM-RL model is composed of descriptive information for requirements such as attribute-value pairs and AND/OR table describing triggering conditions or definitions of corresponding model elements. On the other hand, the Lustre language is designed for efficient simulation and execution of the behavior of reactive systems. Thus, in fact, it is impossible to exactly translate SpecTRM-RL into Lustre one-to-one. Instead, we must focus on the cautious translation of its crucial properties and requirements which are necessarily considered at safety-critical systems.

A. Data Types

SpecTRM-RL supports Boolean, integers, reals, durations,

Table 3: Translation of data types

Type_Decl ::= BOOLEAN | INTEGER | REAL | Duration | Enumerated Duration ::= Integer_Number (NANOSECONDS | MICROSECONDS | MILLISECONDS | SECONDS | MINUTES | HOURS) Enumerated ::= '{' ID { ',' ID }* '}
Type_Def ::= {ID {',' ID}* ':' Type_Decl ';'}+ Type_Decl ::= BOOL | INTEGER | REAL

and enumerated sets. Boolean types are dealt with entirely in the context of matching values in AND/OR tables. The other four types are used as input types, in functions, or in table expressions. Lustre supports integers, Booleans, and reals. For composite types, it supports tuples, records, and arrays. Thus, except for enumerated and durations, the other types can be translated to Lustre codes as they are (See Table 3). In case of enumerated types, we sequentially assign the integers to each element from 0. Durations represent a specific length of time; so duration literal constants can be written in units of nanoseconds, microseconds, milliseconds, seconds, minutes, or hours. Thus, duration constants cannot be negative values. Durations in SpecTRM-RL are translated to positive integer types in Lustre.

B. Model Elements

In model elements, the name of each model element is represented as each node name in Lustre except for mode elements. Below attribute-value pairs, there is an AND/OR table which describes triggering conditions or definitions of each element. These tables are represented as 'If…then…else' statements in Lustre. Note that the length of all variables in Lustre must not be longer than 64 characters because variables larger than 64 characters cause an error when we try to simulate the Lustre codes with 'lux' commands even though they do not cause an error when compiling them. In addition to these common rules, there are more specific rules for each model element.

Outputs

There are two types of output model elements: output command and display output. Regardless of the type, each name of the output elements is represented as a node name in Lustre. The triggering condition table is represented as an 'If…then…else' statement and each condition is represented as a conditional expression with 'AND' relationship in that statement. On the other hand, each column describing 'True', 'False', or 'Optional (*)' is connected with an 'OR' relationship in the conditional statement. We only take 'True' and 'False' conditions into account when composing conditional statements. In particular, the message contents table below the triggering condition table is translated to return values and variables respectively (See Table 4).

Lastly, after finishing translation of each output element, we must create an additional node in Lustre so that we can assemble the output nodes which have already been translated. This final node is named after the project name in SpecTRM. All input parameters and return values of each output node must be placed into the input parameters and return values of this additional node respectively.

Table 4: Output element translation (Part)

Output ::= Output_Type **Output_Name** Destination_Attribute {General_Attribute}*Triggering_Condition_Table Message_Contents_Table Output_Type ::= OUTPUT COMMAND | DISPLAY OUTPUT Output_Name ::= ID Destination_Attribute ::= DESTINATION ':' ID General_Attribute ::= ID ':' TEXT Triggering_Condition_Table ::= {(Expression Boolean_Value)}+ Message_Contents_Table ::= {(Field Value)}+
NODE **Output_Name** '(' {Argument_Decl}+ ')' RETURNS '(' {Input_Return_Argument_Decl}+ ')' ';' [VAR {Variable_Decl}+] LET **Return_Variable** '= ' If_Then_Else ';' TEL Argument_Decl ::= {ID {',' ID}* ':' Type_Decl ';'}+ | ID {',' ID}* ':' Type_Decl Variable_Decl ::= {ID {',' ID}* ':' Type_Decl ';'}+ Type_Decl ::= INT | REAL | BOOL Output_Name ::= ID Return_Variable ::= ID If_Then_Else ::= IF Boolean_Expression THEN {Function_Def}+ ELSE {Function_Def}+ Boolean_Expression ::= And_Expression OR And_Expression And_Expression ::= Not_Expression AND Not_Expression Not_Expression ::= [NOT] (Expression | '(' Boolean_Expression ')') Function_Def ::= If_Then_Else | Output_Return_Value Output_Return_Value ::= Integer_Number | Real_Number Input_Return_Argument_Decl ::= **Return_Variable** ':' (INT | REAL)

Modes

Unlike other model elements whose names are directly represented as node names, each transition name of mode elements must be respectively represented as each node name in Lustre because not only is each transition of modes, in most cases, referred to individually by other nodes, but also each condition in a transition of modes frequently refers to other transitions. Since each transition name is translated to each node in Lustre, the return type of each node must always be of a Boolean type. In particular, each transition name must show up only once in the model and cannot be the same as the name of a macro in the same model. Other rules regarding the AND/OR table in modes exactly follow the rules for outputs.

States

The name of each state element is represented as each node name in the Lustre language. In general, the translation rules of state elements exactly follow the rules for outputs except for the return values. In each state model element, there are several transitions in definition section. Each transition is assigned integer values from 0 to the number of transitions. Because all states require a transition to 'unknown', 0 is always assigned to the 'unknown' transition in every state element. Then, if all conditions of a transition are satisfied, this node returns the corresponding number assigned initially as a return value; so the return type of state element must always be an integer type.

Macros

Macros are, in general, provided to increase readability and understandability. Macros take a piece of an AND/OR table from another part of the model and endow it with a name; so the definition of macros consists of a single AND/OR table evaluating to true or false. Thus, the return type of macros must always be a Boolean type.

Inputs

Inputs have a type of integer, real, duration, or enumeration. Translation of these types follows the data type translation rules mentioned above. In particular, input elements have several critical translation issues. I will describe these issues in detail later in section IV.

Functions

A function definition language includes an if-then-else block, while-repeat loop, assignment, and a return statement. In addition, all the operations, predicates, and primitive available in AND/OR tables are also available inside SpecTRM-RL functions except only for time primitives. In the case of if-then-else block, assignment, and a return statement, they can be translated into Lustre as they are. However, since Lustre does not support the function of loops, while-repeat loops in a SpecTRM-RL function language cannot be relatively translated into Lustre.

C. Expressions

The majority of SpecTRM-RL's expressive power results from the AND/OR tables that define most of the behavior of the components [11]. In those AND/OR tables, there are many expression templates frequently used in SpecTRM-RL. In this section, I directly translate those expressions to Lustre codes. Note that I present only a part of all expressions and translated codes, and provide directly translated Lustre codes, not BNF expressions, for understandability (See Table 5).

Table 5: Translation of Expressions

SpecTRM-RL Expression	Lustre codes
Previous Value of InputName **is** InputValue	pre(InputName) = InputValue
InputName **N Transition Ago is** InputValueName	Init → pre(Init → pre(· · · → pre(InputName))) = InputValueName
InputName **was Received**	node *Received*(Variable: int; Flag: bool) returns(Is_RCVD: bool); let Is_RCVD = if(Flag)and((Variable=value) then true else false; tel
Time Since InputName **was Last Received**	node *Time_Since*(Variable: int; Flag: bool) returns (Diff: int; NoValue_Flag: bool); var CLK, LAST_CLK: int; Is_RCVD, Ever_RCVD: bool; let CLK = 0 → (pre(CLK) + 1); Is_RCVD = if(Flag)and((Variable>= -Val) or(Variable <= Val)) then true else false; LAST_CLK = 0 → if Is_RCVD then CLK else pre(LAST_CLK); Ever_RCVD=if(Is_RCVDorpre(Is_RCVD) then true else false; (NoValue_Flag, Diff) = if Ever_RCVD then (false, CLK - LAST_CLK) else (true, 0); tel

IV. Translation Issues

This section describes several major translation issues especially for safety-critical systems and their approaches, which I proposed.

A. Data synchronization

Most systems, including safety critical systems, in general, have at least several or lots of input data. At a certain point, the system may receive some input data, and may not be able to receive the rest. In other words, the system cannot receive all the input data at the same time. With SpecTRM-RL, we can specify theses properties because it is a modeling language to describe the behavior of system components. However, Lustre is a functional language operating on stream which is a finite or infinite sequence of values. Thus, at the n^{th} execution cycle of the program, all the involved streams take their n^{th} value. But this assumes that the system always receives all input data at the same time.

Approach

When input model elements in SpecTRM-RL are translated into Lustre codes, each input node in Lustre retains its own flag whose data type is Boolean so as to determine whether the corresponding input data was received or not.

B. Deadlock

Each mode transition in the control mode must be translated into each node respectively in Lustre because each mode transition often refers to another mode transition as well as to other model elements. In particular, if one of conditions in a mode transition definition table refers to its own mode transition, this causes deadlock.

Approach

If we initialize the value of the condition, and then make the condition refer to the previous value of its own transition mode, we may be able to solve this serious problem. However, this cannot be a complete solution because the current value always refers to the one-step previous value. This is one of my future projects.

C. Obsolete data

Every input model element requires an obsolete transition. Many accidents have been caused by such software that operated on data which is too old to be reliable. This obsolete transition forces consideration of how long data is valid and what to do when no valid data has been received for longer than that time, or when valid data is never received. So, even though the system has received some input data from sensors measuring values or from operator controls, the system must check whether the received data is valid or not.

Approach

Each translated input node in Lustre contains an extra flag (Obsolete_Flag) as a return parameter which shows if the data was obsolete. Thus, if the input data causes an obsolete transition, the flag must be set as a true in order to notify the system of the input data status.

D. Nondeterminism

Nondeterminism occurs when conditions in the SpecTRM-RL triggering condition tables are not mutually exclusive. Since SpecTRM-RL is just a requirement modeling language to describe requirements, it does not support checking for nondeterministic decisions.

Approach

When nondeterminism occurs in SpecTRM-RL, formal semantics such as BNF of SpecTRM-RL mandates a nondeterministic choice. So, we assume that when we meet a nondeterministic choice in SpecTRM-RL, the mandatory choice is the first satisfied condition. In fact, we may be able to try to solve this problem with different approaches such that we add additional conditions. But this approach can make the translated codes more complex.

E. Other issues

In addition to these major issues mentioned above, there are some more minor issues: System start primitive, time expressions related with a special value – NoValue, time granularity, and so on. I have also proposed approaches for these issues.

V. The Experiment

In this section, I translate two real SpecTRM-RL models to Lustre codes by using the rules I proposed, and then perform black box testing in order to verify the results.

A. Translation

I used two practical SpecTRM-RL models (Altitude Switch and Cruise Control) provided by SpecTRM (Version 1.0.50), and translated these models to Lustre codes respectively. The translated LOC (Line Of Code) was 974 lines and 1470 lines each.

B. Testing and results

SpecTRM-RL is a requirement modeling language for describing requirements; so we cannot use the SpecTRM-RL model itself for testing. Instead, SpecTRM supports simulation to make it possible to visualize how the component will behave before the expensive steps of design and implementation (See Fig 2). SpecTRM simulates the behavior of the component by executing the requirements specification. By using this simulation, we can check the output values and state values of the SpecTRM-RL models according to their input values. So we can use these values for comparing with the output values from the Lustre codes.

I prepared two test files for each program, and each test file contains 100 test cases for input values. I operated SpecTRM simulation for 100 seconds with those test case files, and then

Table 6: A sample of test results

Sec	Inputs							Outputs	
	DAT Signal	DAT status	DAO signal	DAO status	AA signal	AA status	GS signal	G cmd	W strobe
0	Obs	Obs	Obs	Obs	Obs	Obs	Obs	N/A	High
1	300	Norm	300	Norm	Above	Valid	Raised	N/A	N/A
2	290	Norm	290	Norm	Above	Valid	Raised	N/A	N/A
3	280	Fail	280	Fail	Above	Valid	Raised	N/A	N/A
7	240	Norm	240	Norm	Below	Valid	Raised	*Lower*	*N/A*
19	200	Fail	200	Fail	Below	Valid	Raised	Raise	*High*
20	200	Fail	200	Fail	Below	Invalid	Low	Raise	*High*
53	0	NCD	0	NCD	Above	Invalid	Low	*Lower*	High
97	-200	Test	-200	Test	Above	Invalid	Low	Raise	High
98	-200	Obs	-200	Obs	Above	Invalid	Low	Raise	High
99	-200	Obs	-200	Obs	Obs	Obs	Obs	Raise	High
100	Obs	Obs	Obs	Obs	Obs	Obs	Obs	*Lower*	*N/A*

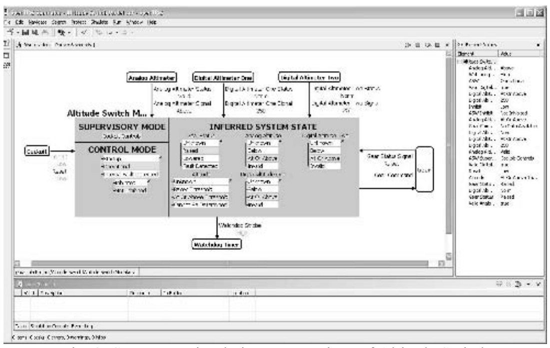


Fig. 2: SpecTRM simulation perspectives of Altitude Switch

compared the output values from the simulation with the output values of Lustre codes. I used SpecTRM version 1.0.50 and Lustre version 4.0.

In the first model (Altitude Switch), 95 out of 100 test cases were equivalent to each other. Table 6 shows a part of the first testing results. The main reasons of the five mismatched results (bold characters) were as follows: time granularity at 7th second, deadlock at 19th and 20th seconds, and nondeterminism at 53rd and 100th seconds. Additionally, at the second testing of this first model, 96 out of 100 test cases were matched with each other.

In the second model (Cruise Control), 92 and 94 out of 100 test cases were the same as each other. The second model testing also showed that the most mismatched cases result from the issues I mentioned in section IV, especially, nondeterminism and deadlock. These results give me explicit future work. Additionally, these Lustre codes were also effectively translated to other analysis codes by using a previously developed translator.

VI. CONCLUSION

SpecTRM-RL is a requirement modeling language for describing the behavior of system components. However, SpecTRM does not support formal verification which plays a very important role in developing safety-critical systems and software. Lustre is a dataflow synchronous language designed for programming reactive systems. Lustre supports verification and analysis as well as code generation. By translating a SpecTRM-RL model into Lustre, we can endow SpecTRM-RL with the features of verification and analysis Lustre originally has.

In this paper, I presented the rules to translate a SpecTRM-RL model into the Lustre language, and also presented an empirical study in which I translated specific SpecTRM-RL models into Lustre with the rules I proposed. The studies showed that the SpecTRM-RL model can be effectively converted into Lustre so that it can support formal verification. In addition, the Lustre codes can be automatically translated into other codes of analysis such as NuSMV, PVS, and SAL by using a previously developed translator. As a result, SpecTRM can be a more powerful toolset for safety-critical software and system development.

The empirical results led me to an explicit research direction and future work. As mentioned before, most of the mismatched cases resulted from the translation issues: especially, deadlock and nondeterminism. In addition, I am also working on finding new approaches to figure out the statements Lustre originally does not support: loops and 'while-repeat' loops. Recently, I have been investigating various ways to further improve the efficiency of the rules.

REFERENCES

[1] Yunja Choi, Mats P.E. Heimdahl, "Model Checking RSML-e Requirements," hase, p. 109, *7th IEEE International Symposium on High Assurance Systems Engineering (HASE'02)*, 2002.

[2] L. Grady, J Howard, and P. Andersson. Safety-Critical Requirements Specification and Analysis Using SpecTRM. *In Proceedings of the 2nd Meeting of the US Software System Safety Working Group*, Feb 2002.

[3] An Introductory Guide to SpecTRM, *http://www.safeware- eng.com/products/ spectrm/SpecTRMTutorial.pdf*

[4] N.Halbwachs, P. Caspi, P. Raymond, and D. Pilaud. The Synchronous Dataflow Programming Language Lustre. *In Proceedings of the IEEE*, Vol. 79, No.9, pp. 1305-20, September 1991.

[5] A Tutorial of Lustre. *http://www-verimag.imag.fr/~halbwach/PS/tutorial.ps*

[6] P. Caspi, D. Pilaud, N. Halbwachs, and J. Plaice. Lustre: a declarative language for programming synchronous systems. *In 14th ACM Symposium on Principles of Programming Language, POPL '87*, Munchen, Jan. 1987.

[7] S. P. Miller and J. Duffy. Autocoding Tools Interim Report. Rockwell Collins, Advanced Technology Center Technical Report in contract NCC-01001, February 2004.

[8] NuSMV 2.3 tutorial. *http://nusmv.irst.itc.it/NuSMV/tutorial/v23/tutorial.pdf*

[9] Sam Owre, John M. Rushby, and Natarajan Shankar. PVS: A Prototype Verification System. *11th International Conference on Automated Deduction,* volume 607 of Lecture Notes in Artificial Intelligence, pages 748--752, Saratoga Springs, New York, June 1992. Springer-Verlag

[10] An Introductory Guide to SpecTRM, *http://www.safeware-eng.com*, Accessed February 14th, 2004.

[11] SpecTRM User Manual, Safeware Engineering, 2003.

[12] J. A. Plaice and N. Halbwachs. Lustre-v2 user's guide and reference manual. *Technical Report SPECTRE L2*, IMAG, Grenoble, Oct. 1987.

[13] Adrian Curic. Implementing Lustre programs on Distributed platforms with Real-Time Constraints, Ph.D.Thesis, Universite Joseph Fourier, Grenoble, France, July 2005.

[14] Jean-Louis Camus. Efficient Development of Avionics Software with DO178B Safety Objectives, *http://www.esterel-technologies.com/files/ AeronauticsHandBook-SC-HB-DO178B-KCG42-DL.pdf*

SECURITY IN INFORMATION SYSTEMS: SOCIOTECHNICAL ASPECTS

Prof. Edison Luiz Gonçalves Fontes, CISM, CISA – GTECH – São Paulo - SP/BR –
edison@pobox.com
Prof. Antonio José Balloni – CenPRA/MCT – Campinas – SP/BR –
antonio.balloni@cenpra.gov.br

Abstract - This work considers the sociotechnical approach that must be addressed when implanting and having maintenance of the information security of an organization. In general, only the technical aspects are considered by the IT professionals. These technical aspects generates in the organization in the day by day, a security process that seems as a set of distant rules and, because of this, the security process (the rules) will not be internalized by the users. To consider sociotechnical aspects means both making a complete approach for the security subject as well increasing the chances for the existence of a continuous process, following and protecting the information resources during the growing stages and moments of difficulties in the organization.

I. INTRODUCTION.

The security in information systems must contemplate not only the technical aspects. The social aspects related to the organization environment and related to the people also have its importance and must be considered.

Due the fact that, historically, the information security began from the technical area of data processing then, the social aspects of the organization and the people have been left of side. Another important fact that must be considered is that even the technical aspects possess a ampler connotation ***[01]***. Furthermore, while security breaches and damage to information systems still come from organizational insiders, security breaches from outside the organization are increasing because firms pursuing electronic commerce are open to outsiders through the Internet. It is difficult for organizations to determine how open or closed they should be to protect themselves. If a system requires too many passwords, authorizations, or levels of security to access information, the system will go unused. Controls that are effective but that do not prevent authorized individuals from using a system are difficult to design.

This work present a approach for information system security that surpass the technical aspect warning the managers from information security as well as the executives of the organizations that the protection of information resources must consider the sociotechnical perspective.

II. SOCIOTECHNICAL SECURITY INFORMATION SYSTEM: DEFINITIONS.

3.1 - Socials concerns:

These are the aspects strongly related to peoples and to the environment where these peoples live and work.

3.2 - Technical concerns:

They are the aspects strongly related to the technology and to the resources of technology.

3.3 - Sociotechnical perspective:

In a sociotechnical perspective the performance of a information system security is optimized when both the technology and the organization mutually adjust to one another until a satisfactory fit is obtained ***[02]***.

III. PLANNING THE SECURITY OF INFORMATION: SOCIOTECHNICAL DESIGN.

For implementing a information security process it is primordial the elaboration of a planning but, what about the information security?

- Information security: whereas information security used to be an arcane, technical topic, even CEO know about it today due to importance of electronic information in running their business. Actually, all business executive now need to understand Internet-based threats and countermeasures and continually fund security work to protect their businesses. At first, when government and industry became aware of the need to secure their information resources, attention was focused almost exclusively on protecting the hardware and data and the term system security was used. This narrow focus was subsequently broadened to include not only hardware and data, but also software, the computer facilities, and personal as well. Today the scope is even broader, to include all types of data. The term information security is used to describe the protection of booth computer and non computer equipment, facilities, data and information from misused by unauthorized parties. This broader definition includes such equipment as copiers and fax machines, and all types of media, including paper documents ***[03]***. These aspects, forcedly, forward to a Sociotechnical design.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 163–166.

- A sociothecnical design is a design to produce information systems that blend technical efficiency with sensitivity to organizational and human needs i.e., a sociotechnical design plan establishes human objectives for the system that lead to increased job satisfaction. Designers set forth separate sets of technical and social design solutions. The social design plans explore different work group structures, allocation of tasks, and the design of individual jobs. The proposed technical solutions are compared with the proposed social solutions. Social and technical solutions are, therefore, a sociotechnical solutions. The alternative that best meets both social and technical objectives is selected for the final design. The resulting sociotechnical design is expected to produce an information system security that blends technical efficiency with sensitivity to organizational and human needs, leading to high job satisfaction. Systems with compatible technical and organizational elements are expected to raise productivity without sacrificing human and social goals ***[04]***.

3.1 – Guidelines to Information Security.

The goal of Information security is to achieve the following objectives: confidentiality, availability and integrity. Confidentiality: the firm seeks to protect its data and information from disclosure to unauthorized persons. Availability: The purpose of the firm's information infrastructure is to make its data and information available to those who are authorized to use it. Integrity: data with an unreduced or unbroken completeness ***[05].***

Next lets consider the main item necessaries when implementing a planning for a information security process.

The security of the information is rich in operational activities and, as function of its weaknesses, we are guided to, immediately, start with technical actions which are, at first, considered as the "most important". However, there is an potential hazard if we are only limited to these operational activities! As the operational activities are an important elements for the organization then, it is very important having an elaborated Information Security Strategic Planning (ISSP) and, this ISSP must be validated by the high administration of the organization, guiding the ways that the projects and activities must follow.

The basic guidelines for implementing the ISSP to be followed are:

a) being aligned with the organization's politics and legislation.

All actions for security of the information must respect the actual state legislation as well the organizational politics.

b) considering the business initiatives.

The most important action for the businesses of the organization is its accomplishment, i.e, the survival from the enterprise. Therefore, for the accomplishment of a viable business, the security must guarantee that the information use in the several initiatives of that business, is happening in a adjusted way. At the same time, an extreme protection can make a nonviable business.

c) defining the structure of a security area.

Should we use ours humans resources for ours projects or, the human resources from others areas? How the scope of the Information System (IS) becomes in the organizational structure? How these and others definitions regarding the area of security need to be predefined?

d) defining the operation form.

Together with the structure definition it is necessary to mount the operation form and scope area of the information security. In general, the computational environment must be contemplated but, depending on the kind of organizations, several subjects as equipment protection, environment and peoples etc, should also be cared for since these subjects are in a gray area.

Following these basic guidelines, a good strategy must be divided into three components: Architecture, Commitment and Protection actions

- *Architecture.*

The security process of the information must follows a viable (possible to be carried out) architecture ***[06]***. In terms of protection approach, this architecture allows a complete vision of a practical architecture ***[07]***.

- *Commitment.*

The user commitment is the support (pillar) for the organizational information security effectiveness.

- *Protection actions.*

Here fits in all the procedures, technical or not, that will drive the protection of the information. Sometimes we are compelled to consider only this approach: protections actions.

This practical example, figure 1, of strategic planning is not a closed rule therefore, it must be applied based on necessaries adaptations for the reality of your organization.

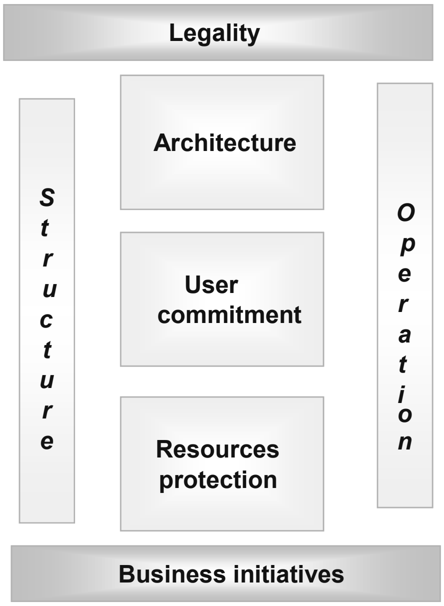


Fig. 1. The strategic planning of information security must have a structured approach and must consider the aspects described in this figure. Many times the IT professional on security information is limited only to the Technical Resources Protection aspects and along the time the process of information security loses its effectiveness as consequence *of this restricted approach.*

4. SOCIAL ASPECTS

5.1 – Regulations

The regulations (politics, norms and procedures) provide the definitions and makes explicit what must be considered as a standard behavior. The peoples must follow these regulations, otherwise they will be breaking the organization coexistence rules.

It is very important for the social environment the existence of these explicit rules.

5.2 – Organizational culture

This social aspect is built along the time. The culture from the organization must be considered when Implanting a Information Security Process and, all the securities controls must be defined so that this culture is respected ***[05]***. On the other side, this does not mean that we have to give up our strategies about information security in benefit of that culture. Therefore, the implantation of a information security process must take into consideration the existing organizational culture. The implantation of a information security process in the army is very different from the implantation of a information security process in an advertising company.

5.3 – Organizational environment.

When a company as an excellent organizational environment this will facilitate the Information Security Implantation Process (ISIP).

A environmental lack of trust between peoples, unfriendly peoples, revolt against the organization and other problems, do not hinder ISIP but, become it very difficult. Therefore, ISIP depends on the environment surrounding the organization that use it.

This environment includes economics conditions, characteristics, principal resources (especially labor), management philosophies, societal mores and other factors. This environment changes constantly and, at the same time, technological advances affect the way ISIP is used.

5.4 – Continuous training process.

Organization with continuous training process has a higher awareness about information security. This process creates an positive environment for the person: a constant growing as professional or as human being. As the people from the organization are considered for a training and other actions, the organizational social environment becomes more positive.

5.5 – Professionalism

Many organizations still possess an amateur relationship with their employees and, sometimes, the same amateur relationship occurs between their employees – employees are a mirror from the organization. A ISIP will be better successful in those organizations where the working relationship is guided by the professionalism. The amateur treatment of the enterprise questions seems to be more strong in the small and medium enterprises (S&M) as well in the familiar companies.

5. TECHNICAL ASPECTS.

5.1 – Update technology.

Today there is an incredible speed in the change and improvement of the technologies and, businesses are ready to use these new technologies. Yet, the information security has to follow up and to reach this new technology and, naturally, to define news ways of controls. Therefore, here we have a reason why information security is always behind the business!. Our goal should be looking for ways of guarantee that the delay time between the information security and de advance in the technology be the less possible or, at least, an acceptable difference that does not put in risk the business.

5.2 – The solution provider.

When an organization adopts a technological solution, the organizations becomes dependent from that solution or dependent from its vendor. Therefore, it is very important that a Service Level Agreement (SLA), a formal written agreement made between two parties, be signed between the parties self interested, guaranteeing the continuity as a solution provider. In the last years, even strong companies has been acquired by others organizations and, in this process occurs the frozen for the solution from the first, compelling the customers to move for the solution from the purchaser organization....

5.3 – Keeping security requirements.

It is not by the fact one organization is using a new technology or solution that the security requirements must be forgotten. Independent from the technological solution, individual identification, accesses controls and registers, the backup data base and business continuity are aspects of information security that always must be considered.

6. CONCLUSION.

The security for a information systems is not a trivial activity since it must consider both the technical aspects and the environmental aspects where the peoples are. Business continuity management is achieved by means of a contingency plan, which is usually divided into sub plans. An emergency plan protects the employees; a backup plan enables the operation to continue even after loss of a computing capability; and the vital records plan ensures that data is not lost.

Each one of these aspects can and must be subdivided. The important is to understand this concept and apply it when in a process of Information Security Implantation and maintenance in an organization, remembering that the goal of Information security is confidentiality, availability and integrity.

References.

[01] – FONTES, Edison. Vivendo a segurança da informação, Editora Sicurezza, SP/Br, pp. 75-102, 115-118, 185-196, 2000.
[02] – BALLONI, Antonio José,. Por que GESITI?, Editora Komedi, pp 17-21, 2006
LAUDON, C. Kenneth and Jane P. Laudon. "MIS" (Activebook), Prentice PEARSON/Hall, pp.14-15, 2004
[03] – MCLEOD, Raymond et al. Management Information System, 9th E Prentice Hall, pp. 209-224, 2004.
[04] – MUNFORD, Enid et al. Computer System in Work Design: The Ethics Method, John Wiley-NY, pp. 1979
[05] - PELTIER, Thomas. Information Security Fundamentals, Auerbach Publications, pp 2005
[06] – FONTES, Edison. Segurança da informação: o usuário faz a diferença!, Edit.Saraiva, SP/Br, pp. 1-18, 130-131 2006.
[07] – SHERWOOD, John. Enterprise Security Architecture, CMP Books, pp. 3-16, 217-283, 2005.

Architecture for Distributed Component Management in Heterogeneous Software Environments

E. Stoyanov, A. MacWilliams, Dieter Roller

***Abstract*—This paper proposes an alternative approach to software management in heterogeneous environments. It targets encapsulation and dependency management of component systems using Stanford Beer's Viable System Model (VSM) as requirements and organizational model. VSM is expressed with meanings of Common Information Model (CIM) extensions that serve as basis for an object-oriented representation of managed components. A control-loop architecture is proposed to facilitate monitoring of heterogeneous component environments using the developed model.**

I. Introduction

Software systems are becoming more and more complex every day. They involve different technologies for component management and communication. Software systems are developed and supported by various vendors, often especially tuned for the specific needs of the customer. In this way customers have the opportunity to select the product and the framework that suites them best for the different specific tasks and problems they might experience.

However, there is a side effect of this development, namely the sometimes difficult management of the deployed software. Reasons for it are the lack of common management instrumentation for tracking of problems, such as functional regression, configuration and deep dependencies between components of different nature. In production environment, where development process depends strongly on the availability of IT assets, stability is factor with a higher priority. This paper proposes elements of modern software engineering that affects the design of software support systems in a way to make them flexible and robust. We introduce a software architecture that adapts Stanfrod Beer's Viable System Model (VSM) with the help of which the platform supports inter-framework dependency tracking and related problems.

II. Heterogeneous Systems

Vendors of software support systems are realizing that extensible functionality is an important aspect of the growing demand of the customer companies utilizing the software. This is visible with the increasing usage of Rich Client Platforms (RCP) based on plug-in frameworks [1] and distributed component systems (DCS) [2]. While RCP and DCS are alone very rich environments, deployment of only one of them is not enough for full satisfaction of the user demand. For example RCP contributes to visual arrangement of graphics, tooling interfaces, and overall functionality, but is not designed for distributed tasks. DCS systems solve this problem with their dedicated design for distributed computing, abstraction of resources (such as databases) and standard approach to inter-operable communication and information exchange. It becomes more evident that development of distributed computing inside an organization demands deployment of additional technologies, such as multi-agent systems (MAS) for knowledge communication[3] and web services (WS) for inter-operable cooperation[4]. These interacting components become dependent in heterogeneous manner, for example a component residing in Enterprise Java Bean(EJB) container may be dependent on another remote component located inside .NET framework. Even if some frameworks provide local version and dependency chains management, the missing formal relations between the frameworks on the management level increases the management effort and respectively support costs.

III. Component Frameworks

Before we propose a model for management of heterogeneous systems we will take a short look at what software components are and how they are organized.

The common definition for a component is that it is a manageable and reusable software element that is portable between containers designed to manage its life-cycle using interfaces specified in a component specification and supported by the container. If we take a look at how currently the modern component technologies wrap the system and functional components it can be concluded that there are very general similarities when it comes to encapsulation and communication principles. For a more practical understanding of the problem we will discuss three modern component technologies, used in RCP and DCS in production-ready systems: Enterprise JavaBeans (EJB), OSGi Framework (OSGi), and Microsoft .NET platform (.NET). Deployment of all three of them serves as an usual setup of client side development tools (.NET and OSGi), server-side transactions and database interaction (EJB) and intra- and inter-organization interaction (EJB and .NET). In order to come closer to the idea of heterogeneous management of components we will introduce the common communication principles used in those component technologies, as well as their organizational hierarchy of containment.

A. Component Communication

Assuming the Architecture Description Language (ADL) vocabulary [5], the components communicate with their supporting infrastructure and with other components using

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 167–172.

interfaces with different *types* of *ports*. A *component interface* may consist of the following ports: *attributes*, *methods*, *event source*, *event sync*. Generally, those can be used in those three cases: when the component communicates with the management framework, when it communicates with other components or by communication with traditional (non-component) software entities, such as external functions and interfaces of classes outside the components.

1) *Communication with the Framework:* Communication with framework is usually performed by the components in the case where they need resources or information about certain states of the system. In the case of OSGi, the components (called bundles) communicate with the framework when there is a need to locate other services, as well as for bundle and services state-change notifications. In the EJB framework, the components communicate when they need to locate other objects, or they need special services as timing. The framework on its own communicates with the components with the help of context objects. Information within context is selectively used by the components. In .NET components are communicating with the framework when they need to retrieve component interfaces.

2) *Communication between components:* Communication between components happens on two levels: semantic (functional) and sequential (execution). Semantic communication is related to the way client interacts functionally with the component on the level of their functional interfaces. The sequential communication is usually handled with the help of the framework and is related to the concrete steps with which the actual information and variety transfer is achieved between the components. This includes marshaling/unmarshaling of data-types, data transfer protocol implementation and type mapping. Both aspects are important, because if one of the communicating sides fails to conform to both aspects, the communication between components is either wrong and may lead to undesirable behavior or is completely broken and possibly leading to subsystem malfunction. Figure 1 expresses semantic communication and execution communication.

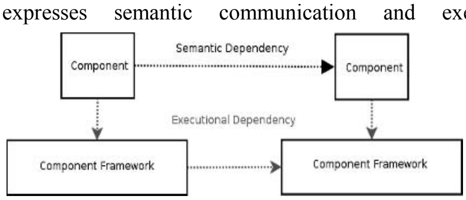


Figure 1. Component Dependencies

To achieve semantic communication the successful executional one has to be present. Another important aspect of component communication is that the component itself is dependent on the underlying framework and essential resources, such as network and database access, are utilized through communication with the upper management levels. Figure 1 illustrates this by the vertical arrows, which can represent queries for resource or communication adapter to achieve the actual communication with other components. That is why an essential point of heterogeneous management is the understanding of the hierarchical grouping and encapsulation of elements.

B. Encapsulation and Hierarchy

While the different component frameworks have different component life-cycle management techniques, there are some very basic common characteristics that deserve attention. Encapsulation is a key principle where components are providing interception points through which the hosting container uses for manipulation of the internal component state and routines for initialization, suspending or destruction. Let us see how the three discussed frameworks define encapsulation. In the J2EE framework the *Enterprise Java Bean* stays on the last level of component management hierarchy inside an J2EE-based system. The upper levels of containment and management are *EJB-Container*, *J2EE Server* and the J2SE *Virtual Machine* as a higher application manager and container. Similar hierarchy is observed in the other component frameworks. For example OSGi defines a component as a set of classes and services in the form of *Bundle*. The OSGi *Framework* acts as a management container for the bundles and stands higher on the hierarchy, followed by the execution environment, typically a JVM. The last level of containment in a *.NET Framework* is the module (DLLs). Assemblies aggregate those *modules* into components that are consumed by applications. Management, such as isolation and automated version handling, are performed by the *CLR* and its dynamic loader. A generalized diagram of encapsulation hierarchy for the three frameworks is shown on Figure 2.

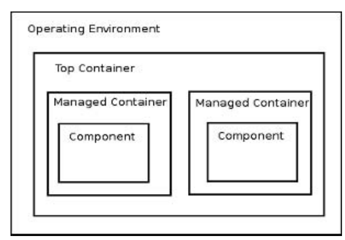


Figure 2. Component Hierarchy

IV. Management of Heterogeneity

There have been many attempts to provide agile software and IT environment in general but few target the problem of evolution support in heterogeneous systems with the accent of management and problem tracking. Our approach to heterogeneous management is based on adaptation of the *Viable System Model* (VSM) [6], a management model that was successfully deployed during the last two decades in different fields of knowledge and organizational management. We re-use the requirements of VSM in software communication and implement a VSM-like model for mapping of component entities. The main advantage of VSM as a template of constructing management systems is the clean definition of communication flow requirements and the task distribution between management entities. One of the main organizational characteristics of VSM is its recursive (fractal-like) model that is the key to cope with complexity management. VSM defines requirements in terms of variety constraints that ensure viability or resistance of the system from perturbations from the environment in which they operate. First implications of VSM into software engineering were described by [7] and developed further by [8, 9] and [10]. In this paper we will discuss a more practical approach to VSM application in the light of its utilization for development of adaptable and evolving software systems.

There are three steps that need to be taken into consideration when adapting VSM to software component systems: the hierarchy of composition, the communication flow and the model of variety. The first two steps will be introduced in this chapter and the used modeling technique is discussed in Chapter 5.

A. VSM-like Compositional Hierarchy

The component containers take care of the different states through which the component has to pass while it exists inside the framework. This transition is known as component life-cycle. In that meaning, every component is being managed by its container, and the container itself can be seen as a manager. On a higher level, the IT elements (servers, switches, etc) need also management and at this point it is primarily function of the human being to keep the system consistent and functional. Current component frameworks are taking care primarily of the states of the component inside their internal existence without taking into consideration the states of other components living outside of the framework, supposedly in other component frameworks. It is also not difficult to see that an organization adopting IT is never fixed to a single centralized framework which results to a natural heterogeneity. That said, there is no holistic model for system stability support, in which the organization of the management units is analyzed and interpreted for the aim of stability.

For better understanding of the material below we introduce the basic concepts of VSM which are explained in detail in [9]. The model is composed of several classes of communicating elements and management units (numbered with 1 to 5): Environment (external for the system), Operations (actuators), Operation Management (*System 1*), Scheduling Manager (*System 2*), Senior Management (*System 3*), Prediction Manager (*System 4*), Policy and System Identity Manager (*System 5*). These elements are connected with *Communication Channels*. System 4 has direct view on the Environment and has the task to predict its possible future state. Starting from the lower level of the Operation, these elements form several homeostatic systems – System 3-1 is a loop cycle keeping track of operation status, System 4-3-1 introduces planning and adaptation to the changing environment. System 5 completes the system with identity (policy) and understanding for the main purpose of existence of the system. System 2 is responsible for synchronization and scheduling of operations. VSM has recursive structure, where System 1 by definition is a whole VSM model (System 5-4-3-2-1) and has the same structure as the system into which it resides. The same rule is applied to all sub-levels of the system. Additionally to the structure, VSM states a set of requirements that assure a proper operation and viability of the system. Most of these requirements are applied to communication flow through the channels, as well as the dynamics of the system.

A VSM-like design may favor the solving of this problem by defining a tiered and self-nested architecture of management and operation dependability which is reflected very comfortably in the way component containers are organized. The key of mapping the containers and components to this structure is the interpretation of the tasks of the individual components. For example, the task of the EJB component is mapped on to the operational unit of VSM, as they are the last level in the hierarchy, where the component container is mapped onto System 1 from VSM. The next iteration on the parent levels shows how the relation Server – Container system is mapped as System 3-1 from the model of VSM. Figure 3 displays the mappings of two iterations in depth Operational-Environment, System 1-Operation and two iterations in depth of System 3-1.

The VSM model includes additionally System 4 and System 5 as an intelligent and policy driven management related to a predictive and proactive behavior. These systems are missing in our model as intelligent tuning is yet topic of research and is immature in the field of software engineering. However, the set of requirements applied to the communication flow (see bellow) is fully applicable to the reduced set of management units in regard to safe system operation. In future developments the note of sub-system 4-5 will be useful as a reference for mapping of predictive and policy driven engines applied in component frameworks.

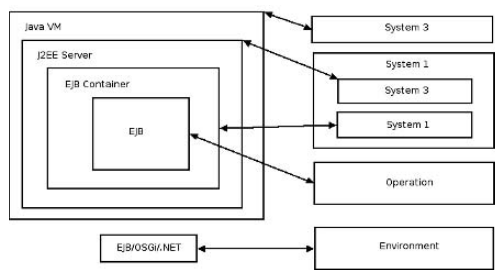


Figure 3. Component Hierarchy Mapping

This point of view and the understanding of operational and management part, mapped onto the functional tasks of the component frameworks help to realize how stability constraints can be applied in heterogeneous manner throughout similarly functioning systems, respectively as discussed *EJB, OSGi,* and *.NET*. These constraints are mainly related to the variety transfer between the groups that are connected with communication channels.

B. Management of Communication Channel Requirements

As described earlier, the communication between components happens in several steps, in most cases dependent from each other. While a component targets semantically the communication interface of another component, the length of the execution dependency to connect both endpoints may vary depending on the environment and the communication mechanism itself. In order to have a common terminology we will use the term of VSM Communication Channel as a way to describe a virtual communication media between two elements, both frameworks and components. When we apply the communication paths between software elements in a component framework mapped onto the VSM compositional model from Figure 3, we are able to monitor and apply the requirements that a software system has to follow for a healthy communication. And because the different frameworks follow the same mechanism we are able to track the whole communication flow from the level of the VSM definition. Then, on the basis of these criteria validation and prevention of problems may be pro-actively controlled in the case System 4-5 exists.

Figure 4 illustrates the structural model of a MCC, which is discussed in details by [10]. The management strategy of the channel manager implements the requirements of VSM for *Requisite Variety* and *Requisite Knowledge*. This composition can be successfully applied in semantic and executional channels. For example a mismatch of the interfaces on the semantic level indicates a wrong update of a component (in case communication in earlier point of time was set up correctly). In this case executional dependencies do not need to be activated. This is important when the system supports redundancy in deployment of components, e.g. two versions of the same component reside in same container.

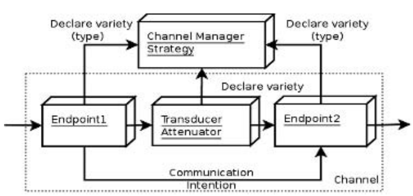


Figure 4. Managed Communication Channel

Then, the manager has the opportunity to select the old semantics and initiate the executional channels to connect the components. Variety exchange between components can be verified and compared using either a dedicated dependency knowledge-base, or, where possible, by comparing the fingerprints of communication interfaces. This flexible channel formation and monitoring facilitates support of stability in the phase of system update and can be applied on any mapped level of software variety to the model of VSM.

V. Application of The Architecture

While software component management until now was not considered to be an issue and so close as a case for special management, there are already management frameworks that fill the gap of distributed management when it comes to monitoring and control of hardware devices and system services. A promising management model in this aspect is the Common Information Model (CIM). It is a platform independent schema and a set of predefined models that is utilized by the Web-Based Enterprise Management (WBEM)[11] framework, both initiatives started and supported by the Distributed Management Task Force (DMTF).

A. Modeling Component Communication with CIM

CIM has a rich set of elements in its model that reflect the way IT infrastructure is organized. The elements that represent managed devices or components are derived from CIM_ManagedElement class. Additionally, CIM supports the notion of Systems (class CIM_System), where they aggregate CIM_ManagedSystemElement objects and provide additional properties and relations to other parts of the system. These classes are abstract and serve as a core of a more detailed CIM models (profiles). This helps to follow a standard management practice across managed environments and help easier traversing of component encapsulation. We will use these features of CIM to construct a VSM-like management model to allow verification of the discussed requirements (see Figure 5). There are four basic classes that we have to model in order to represent a VSM inside a WBEM management framework.

- Environment – represents the variety of IT elements with which a component or a system may interact
- Operation – the active elements in the system through which interaction with environment occurs
- Manager – the class of management units that form the VSM system homeostats
- Channel – the communication medium between elements of the system (Operation-Manager, Manager-Manager, Environment-Manager, Operation-Environment)

We implement these classes as shown on Table 1. The prefix "*AC_"* at the beginning of every class is just an internal notation used to indicate classes belonging to our Autonomic Computing research effort.

VSM Element	***CIM Class***	***Inherits***
Environment	CIM_ManagedSystemElemen t	CIM_ManagedElemen t
Operation	AC_Operation	CIM_ManagedSystem Element
Manager	AC_Manager	AC_System
Channel	AC_Channel	AC_Operation
VSM System	AC_System	CIM_System

Table 1: VSM-CIM Binding

The most important thing to start with is the definition of the abstract notion of VSM variety. In IT and software systems this can be anything that has communication interface. That is why we define the VSM variety to be all subclasses of *CIM_ManagedSystemElement* class, thus we use its predefined semantics to refer to all other elements in the model that reside into a predefined system. The operational units are derivatives of it also and are visible as class *AC_Operation*. They are the lowest level of the model and are responsible for the interaction with the environment. According to VSM the operational units communicate with the environment and the management units using channels, so this is next important definition we have to clarify. *AC_Channel* is a class derived from *AC_Operation* and serves as a definition of dependencies on a higher level than the dependency associations in CIM.

In a VSM system the four classes communicate with each other using channels. The different types of channels a VSM system may have are:

- Manager to manager – used in systems 5-4-3-1
- Manager to operation – used in system 1
- Manager to environment – used in system 4
- Operation to environment – used on operation level
- Operation to operation – operations communicate with each other if necessary

In order to define these types of communication, we use CIM-derived associations between the classes. Associations are important for every CIM model because they define semantics between elements. On the base of these semantics a system manager later may query and traverse relations between instances of objects. For this purpose we associate the AC_Channel class with the communicating elements. Table 2 shows the relations used to express channel communication within the system.

Channel Type	***Associations***
Manager-Manager	AC_ManagerToManagerOverChannel
Manager-Operation	AC_ManagerToOperationOverChannel
Manager-Environment	Inherited AC_ElementToElementOverChannel
Operation-Environment	
Operation-Operation	AC_OperationToOperationOverChannel

Table 2: VSM-CIM Associations

Channels have associations to every possible entity in the model – Operations, Management, Environment. It allows an Operation to connect itself to the outside world of CIM definitions through the association *AC_ElementToElementOverChannel*. These associations help in the definition of the input and output variety for a channel and on a later stage to verify its requisite variety

B. Modeling The Organizational Hierarchy

The important aspect of management recursion is reflected in the way *AC_System* and *AC_Manager* are constructed. *AC_System* keeps the original semantic of *CIM_System* but extends its meaning by being a base class of the management unit *AC_Management*. At the same time it aggregates management units. In case we want to add System 4 to the system it needs to communicate with the environment, that is any other element in the system. This is facilitated by inheriting association *AC_ElementToElementOverChannel*. System 1, by definition is represented as complete VSM system mapped to *AC_Management*. This fully satisfies the recursive model of VSM. The managers in a VSM-like structure will be able to represent the same structural organization on every sub-level of recursion and communicate with upper and sub-levels using communication channels.

C. Connecting the Models

The last missing component for adaptation of the VSM-like structure we presented until now are the mapping relations between existing management models in the CIM repository of the WBEM framework. It is not realistic that development of new systems can be developed only in the frames of the proposed model, even more, the existing models will have to be re-designed from scratch. That is why there are three more associations serving as mapping relations between existing elements in the CIM model and the newly defined VSM model. The necessary mappings are:

- Framework/Application as System – register existing WBEM managed applications or component frameworks with the VSM model
- Connector/Element as Channel – register an instance, or component acting as connector between two elements as a channel
- Element as Operation – register a CIM instance as operation

The management provider that handles the VSM structures will take into account the mappings and treat the external elements accordingly. Table 3 shows the mapping associations that can be used with CIM-based managed elements.

Mapping	***Association***
Framework or application as VSM System	AC_CIMElementAsManager
Connector or element as channel	AC_CIMElementAsChannel
Element as operation	AC_CIMElementAsOperation

Table 3: Mapping Associations

D. Management Control Loop

Until now we discussed the modeling of the relations between components in heterogeneous environment together with the VSM requirements for stability. We have used CIM for modeling and as its complimentary architecture WBEM provides the management facilities for monitoring, notification and management of the components. This chapter explains how to construct a management control loop for monitoring of cross-platform component dependencies and utilize the proposed organizational and communication model. Figure 6 shows the all-in-one model of the discussed until now topics. The model is stored in a CIM repository managed by the CIM Operation Manager (CIM-OM).

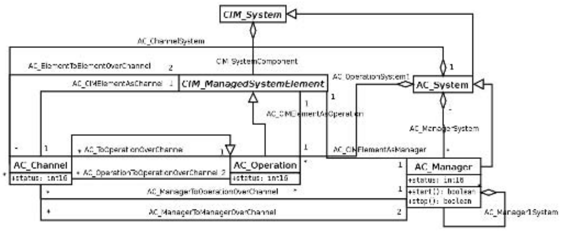


Figure 5. VSM-CIM Model

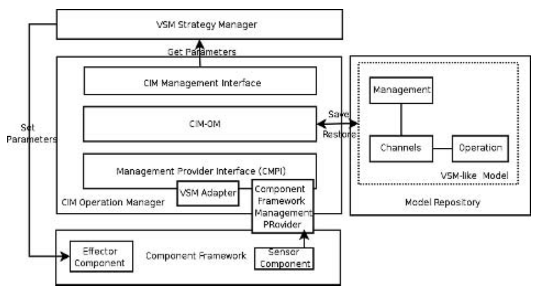


Figure 6. Management Control Loop

The framework defines an abstraction layer through which specific management providers can be hooked. In our example the management provider reflects the component structure in the managed framework and creates instances of the classes from the VSM model. They become available to a monitoring and active manager that follows the requirements of VSM. It closes the control loop with management operations to the framework accessible through effector component.

Effector components are simple management adapters that give access to the framework resources. Such are already available integrated into most frameworks (for example Java Management Extensions interfaces for EJB and OSGi). The manager's goal on the top of the diagram is to analyze the communication and construct valid executional channels to satisfy the semantic communication (see chapter 3). The control loop is in the position to detect inconsistencies according to the VSM model and to create problem reports with precise specification of problem origin and level of location.

VI. DEVELOPMENT AND EXPERIENCE WITH THE FRAMEWORK

In the currently developed framework, we implement adapters and mapping for OSGi and EJB with the help of JMX Connectors and adapters communicating over CORBA IIOP. This gives the additional advantage of completely distributed architecture.

Although the approach provides flexible management, the framework introduces difficulties in integration of new component types inside the framework. Their modeling takes time. The model however is flexible enough to allow for self-diagnosis and auto-discovery of problems related to the management framework itself. A self-describing CIM model mapped onto the internal VSM representation will facilitate this and it is the next step in the development of the approach.

VII. CONCLUSION

We have presented the different aspects of heterogeneity in modern software systems and proposed management architecture for distributed heterogeneous environment. The architecture provides the needed monitoring facilities for heterogeneous consistency tracking, composition and communication of component frameworks. Main advantages of this approach are the natural open system construction and abstract definition of requirements for communication that can be applied to a wide range of frameworks and processes.

Further development of the research is dedicated to the integrated self-monitoring of the framework and clear specification of the steps throughout the adaptation of new component frameworks.

REFERENCES

[1] Stevens, G., Quaisser, G., Klann, M., Breaking it up: an Industrial Case Study of Component-Based Tailorable Software Design, Springer, 2005

[2] Mezini,M., Seiter, L., Lieberherr, K., Component integration with pluggable composite adapters. In Software Architectures and Component Technology. Kluwer, 2000.

[3] Dalakakis, S., Stoyanov, E., Roller, D.: A Retrival Agent Architecture for Rapid Product Development, in: Perspectives from Europe and Asia on Engineering Design and Manufacture, X.-T. Yan, Ch-Y. Jiang, N. P. Juster, eds., Kluwer Academic Publishers, 2004, pp. 41-58

[4] Liu, D., Peng, J., Law, K.H. , Wiederhold, G., Sriram, R.D. : Composition of engineering web services with distributed data-flows and computations. Advanced Engineering Informatics 19(1): 25-42 (2005)

[5] Garlan,D., Robert, T., David Wile,D.,. Acme: Architectural Description of Component-Based Systems. In Gary T. Leavens and Murali Sitaraman editors, Foundations of Component-Based Systems, Pages 47-68, Cambridge University Press, 2000

[6] Beer, .S, Cybernetics and Management, English Universities P., 1967

[7] Herring, C. and S. Kaplan. The Viable System Model for Software. in 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI'2000)

[8] Stoyanov, E., Wischy M., Roller, D., "Using Managed Communication Channels in Software Components", Proceedings of ACM Conference on Computing Frontiers, Ischia, Italy, session: Reconfigurable and autonomic computing, ISBN:1-59593-302-6,2ßß6, pp 177-186

[9] Stoyanov E., MacWilliams A., Roller D., "Distributed Software Maintenance Using an Autonomic System Management Approach based on the Viable System Model," icas, p. 58, International Conference on Autonomic and Autonomous Systems (ICAS'06), 2006.

[10] Bustard, D.,Sterritt, R.,Taleb-Bendiab,A.,Laws,A., Randles,R., Keenan,F., Towards a Systemic Approach to Autonomic Systems Engineering, Proceedings of 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05) , pp. 465-472

[11] Distributed Management Task Force (DMTF), Common Management Model (CIM) Standards, http://www.dmtf.org/standards/cim

Assistive Technologies for Physically Handicapped Persons

Emir Skejić
Faculty of Electrical Engineering
University of Tuzla
Bosnia and Herzegovina
e-mail: emir.skejic@untz.ba
&
Marijan Đurek
Faculty of Electrical Engineering and Computing
University of Zagreb
Croatia
e-mail: marijan.djurek@fer.hr

Abstract. People with physical disabilities have limited capabilities in moving, performing manual tasks and taking participation in some life activities. Muscular dystrophy, multiple sclerosis, spinal cord injuries, head injuries, amputations, arthritis, etc., are some causes of physical disabilities. Since that physical impairments can significantly make tedious, even in some cases quite disable using of information technology, it is required to make adaptations that will be enabling the full communication with computers by persons with this kind of disabilities. In this article, some issues about communication between computers and persons with physical impairments are discussed, and are presented a review of available assistive computer technology that make this communication possible. As information education is fundamental education in modern society, the special accent is given to the communication problematic between physically handicapped persons and computers.

Keywords. Physical Disabilities, Assistive Technology, Orthopedic Disabilities, Alternative Devices

I INTRODUCTION

A wide variety of disabilities result in physical impairments, and can range in severity from limitations of stamina to paralysis [1][2]. Physical disabilities can be present at birth, while others are the result of illness or injury. Quadriplegia refers to the loss of function in arms, legs and trunk areas. Individuals with quadriplegia have limited use of their arms and hands and little or no use of their legs. Many require motorized wheelchairs. Arthritis causes inflammation in the body's joints, resulting in pain and mobility difficulties. Back disorders hamper the individual's ability to sit, stand, walk, bend or carry objects easily. Cerebral palsy is the result of brain damage before or shortly after birth; it may result in speech difficulties, walking problems, spasms and lack of muscle coordination.

TABLE I

INCIDENCE OF SOME COMMON ORTHOPEDIC IMPAIRMENTS IN USA

Impairments	**Incidence**
Spina Bifida	6 per 100.000 births
Spinal Cord Injury	approx. 11.000 cases per year
Muscular Dystrophy	1 in 651.450 or 417 people in USA
Native / Acquired Limb Deficiencies	1 in 4.400 people annually
Arthrogryposis	1 in 3.000 live births

Each physical disability results in different levels of physical difficulties, and individuals vary in the way the disability affects them. Physical disabilities can either be permanent or temporary but will affect how an individual accesses the CAP site and the computer workstation. For example, an elevator or ramp provides access to spaces when a staircase is insurmountable for someone who uses a wheelchair. Similarly, specialized hardware or software, called assistive or adaptive technology, allows people with physical impairments to use computers. These tools allow a person with limited, uncontrollable, or no hand or arm movement to successfully perform in school and job settings. Adaptive technology can allow a person with a physical impairment to use all of the capabilities of a computer. Special input devices for users with physical disabilities will depend on the user's specific impairment; numerous assisting devices are available. The potential for great benefit to people with different kind of disabilities is one of the unfolding gifts of computing.

Given the growth of disabled population expected in the future, the challenge for designers of information systems are the issues of both accessibility and functionality. There are a wide range of available assistive technology solutions on the today's computer market. In this paper is tried to present the

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 173–178.

comprehensive review of adaptive computer equipment and to point out the innovative high-tech solutions.

II COMPUTER ACCESS FOR PEOPLE WITH PHYSICAL DISABILITIES

Common supports and accommodations for people with physical disabilities in the CAP sites might include accessible parking, priority registration, accessible facilities, lab or computer assistants, simple adaptive computer technologies (such as key guards), consideration of workstation set-up and note takers during lessons.

A variety of issues must be considered before addressing access to the computer for people with physical impairments. These include seating and posture, work surface, lighting, temperature, vibrations, noise, ventilation, keyboards (information input) and mouse access, monitor (information output) and accessories.

Computer access for people with physical disabilities may be achieved through:

- keyboard adaptations,
- alternative keyboards,
- an expanded keyboard,
- a mini keyboard,
- mouse alternatives, and
- assistive technology software.

Simple solutions include the modification of key repeat rates and sequential keystroke selection. Keyboard macros allow the user to assign a few keystrokes to perform functions that would normally take multiple keystrokes. Word prediction software limits the number of keystrokes required to enter words and phrases.

Keyboard adaptations can be made with hardware or software. Hardware includes key guards and key locks. Software adaptations include Easy Access (Mac) and Access DOS (IBM) alternative keyboards, including PowerPad, Big Keys Unicorn Expanded Keyboard, Intellikeys, TouchWindow and on-screen keyboards, Tash Minikeyboard, KeyLargo, chordic keyboards, Braille keyboards, and so on. Some alternative keyboards plug into the serial ports of any computer (Intellikeys). Other devices (Unicorn Expanded Keyboard) require additional equipment.

More involved physical disabilities require alternative input, including:

- the TongueTouch Keypad,
- single-switch on-screen keyboard access with scanning,
- single-switch access using Morse code,
- voice recognition software,
- head mouse/head master and pointing devices,
- JOUSE - a joystick-operated mouse controlled with the mouth,
- voice input,
- eye gaze technology,
- head wands or Sip N' Puff, and
- on-screen keyboards with regular or alternative mouse access (track pads, joysticks, trackballs).

III ASSISTIVE TECHNOLOGIES FOR PERSONS WITH ORTHOPEDIC DISABILITIES

Orthopedic disabilities can result from many causes such as accidents, strokes, birth defects, viral infections and neurological disorders. The range of physical impairment that result from such disabilities is enormously varied. This section addresses the access consideration of persons who can press all of the keys on a standard computer keyboard using either their fingers, a hand, head or mouthheld pointing device, toes or other body extremity [4][5][6].

For persons with disabilities that affect the upper body, productive use of computers should address three critical issues: keyboard positioning, keyboard access and typing speed.

III.A Keyboard Positioning

The keyboard can be the biggest obstacle to computing for a person with a physical impairment. Fortunately, those who lack the dexterity or range of motion necessary to operate a standard keyboard have a wide range of options from which to choose. Correct keyboard positioning will allow persons with moderate levels of orthopedic disability to minimize physical exertion and thus reduce fatigue. Properly positioned keyboards also help to decrease the spasticity and resultant keyboarding errors that occur from straining to reach portions of the keyboard. Repositioning the keyboard to the floor can allow someone to use his feet instead of his hands for typing.

III.B Keyboard Access

Assistive technologies that provide keyboard access are vitally important. The multiple keystroke commands common to many computer applications can be an obstacle for persons with virtually any degree of orthopedic disability. How, for example, can a one-handed typist or headstick user, hold down a key on the left-hand side of the keyboard while simultaneously pressing another key on the right-hand side of the keyboard?

III.C Specialized Adaptations to Control Keyboard Functions

One of the most useful access tools available for individuals with orthopedic disabilities are programs that provide control of keyboard operation. If physical disabled computer users find that using the standard keyboard works for them but that some aspects are slowing them down, then keyboard modification software may be just what they need. For many disabled persons, use of such an adaptation may be all that is required to gain computer access. Programs of this type should meet the criteria that follow.

Such assistive tools should be utility programs that can be easily loaded into the computer. Keyboard modification software allows users to keep using their ordinary keyboard, but provides some vital assistance just where it's needed. In this way, the disabled computer user can readily move from computer to computer without being dependent on specialized hardware modifications attached to a single computer system. Before purchasing a complex keyboard option, evaluate the accessibility features that are built-in to current popular operating systems. For instance, the Accessibility Options control panel in current versions of Microsoft Windows™ contains a variety of settings that make a standard keyboard easier to use. The Macintosh operating systems have similar features in the Easy Access control panel.

The automatic key repeat function common to many computer keyboards can be a serious obstacle to many physically disabled individuals. Limitations in fine motor control may take the quick release of keys difficult or impossible. Keyboard control programs should be capable of turning off or modifying the key repeat function.

Simultaneous, multiple key commands create a significant barrier for one-handed or touch-stick computer users. Many widely used computer programs make extensive use of special keys on the keyboard such as the Ctrl and Alt keys to carry out commands. The complex feats of manual dexterity required by many programs tax the skills of individuals with a complete set of digits; without keyboard control programs, such programs are virtually inaccessible to persons with significant orthopedic disabilities. Keyboard programs should be capable of electronically "latching down" the Ctrl, Alt and Shift keys individually or in combination. The program should provide an automatic release feature to "unlatch" these special keys after a second, nonspecial key has been pressed.

For moderately orthopedically disabled computer users, one of the most frustrating aspects of using a physical keyboard is the unintentional pressing of keys. Individuals with limited fine motor control often brush unwanted keys with a protruding finger or misdirected pointing device in the process of pressing the desired key. A great deal of time is thus spent erasing unwanted characters from the computer display. The traditional approach to solving this problem has been through the use of key guards. A keyguard is a sheet of flat metal or plastic designed to prevent accidental keystrokes and to provide a convenient place to rest hands when doing data entry. The guard has finger-sized holes, one for each key. In order to press a key, user must insert his/her fingers into one of the holes on the guard. Also, if user is using a mouth stick or head stick to type, the guard can help him/her to increase accuracy and prevent typos. The benefits of a keyguard also can enjoy nondisabled computer users. This same result can be attained using keyboard control programs which, in effect, tell the operating system the amount of time the keys on the keyboard must be held down before sending a letter to the screen buffer. By introducing a very small delay factor, the great majority of accidental keystrokes can be eliminated without significantly reducing typing speed.

In order to provide the disabled computer user with complete access to the full range of commercially available software, the keyboard control program must not interfere with the simultaneous operation of other programs.

The keyboard control program should provide a method for program start-up using a previously chosen selection of keyboard control options.

III.D Enhancing Typing Speed

For individuals with mild or moderate orthopedic disabilities, it is a relative simple task to provide greatly improved access to the computer keyboard. Persons, whose disabilities prevent them from typing at a rate greater than 10 to 12 words per minute, may also require assistive technologies that enhance the rate of text production.

One solution might be word completion programs, which make surprisingly accurate predictions about word choice, while a sentence is actually being written. Using a history of the user's word frequency patterns and word choices preferences, such systems can predict the completion of a word being written, based on its first, second or third letter. The user is shown a list of likely choices and may elect to complete the word or phrase by pressing a single key. Such systems also automatically manage the tasks of inserting the correct number of spaces after punctuation marks and beginning each new sentence with a capital letter. Some word prediction software automatically collects new words as they are used, and consider a person's common vocabulary when predicting in the future.

Word prediction programs should operate transparently with commercial software applications, allow for "on-the-fly" addition of new words and phrases, and constantly adjust word usage frequency tables to enhance the speed and accuracy of word prediction. Word prediction is often used with a virtual keyboard to increase accuracy and typing speed. For those who type much faster than 13-15 words per minute, however, use of word prediction can actually decrease typing speed, because the user is required to look in two places – the keyboard and the screen.

Spell check and correction programs that continuously monitor spelling and offer to correct errors automatically as a document is being written can substantionally reduce the amount of time ordinarily required for such tasks. The kinds of spelling errors that sometimes occur as a result of miskeying due to limited fine motor control can be instantly corrected.

Within the last ten years, great strides have been made in the development of fast, accurate, large vocabulary speech recognition systems for microcomputers. Speech recognition systems can serve as a supplement, or even a replacement, for data entry on a physical keyboard. However, it is required to be carefully with speech input systems because they have severe limitations about distinguishing between similar sounding words or phrases so they are likely to make mistakes. Therefore, for the system to be effective, careful consideration of the person and the task involved is required.

IV TECHNOLOGIES FOR STUDENTS WITH SEVERE PHYSICAL DISABILITIES

Students with severe physical disabilities are a heterogeneous group. For some, mobility is the greatest barrier they face. For others, caring for their personal needs is a tremendous challenge. Still others face overwhelming obstacles in communication. The data indicate that approximately 48.000 students with orthopedic impairments in the United States [3] were served in the public school system for the 1995 year, slightly more than 1% of all students with disabilities who are currently receiving special education services.

Orthopedic disabilities can significantly impact a student's ability to function at school. These can be divided into siy categories and may result in a number of difficulties, socially, emotionally, psychologically and physically. These categories are as follows:

- Poor motor skills
- Restricted language
- Lack of experience
- Individual factors
- Psychological factors
- Ineffective learning environment

Fortunately, a variety of new technologies have been developed to help individuals with physical disabilities overcome their challenges and function well in school, work, and home environments. These innovative assistive technologies are readily available and extremely functional. Following are descriptions of several computing tools that have been effectively used by individuals with physical impairments. This list is not exhaustive and should not limit the person with a physical impairment or the adaptive technology practitioner from trying other approaches.

IV.A Alternative Input Devices for Students with Physical Disabilities

Switches

The keyboard doesn't have to be on the desk to be useful. Users can display a picture of the keyboard on the computer screen using an on-screen software package. After that, it is possible to control the on-screen keyboard using single or multiple switches. When the desired key is illuminated or highlighted, users can select it by clicking a switch. Switches control the flow of electrical power to a device that the user wants to turn on or off. There are a variety of input methods that rely on switches. Scanning and Morse code are two of the most popular. Switches can be activated by almost any part of the body a person is able to voluntarily and reliable control – for example, switches are available that can be activated by the use of an arm, hand, finger, leg, foot, head or chin. They also may be controlled by less obvious movements of the eyebrow, or the rib cage with access through controlled breathing. While the movement does not have to be big, it must be controllable and reliable, and often considerable training is required before the use of the switch is reliable. Morse code is a more direct method of control than scanning and with practice can be a very efficient input methods.

Alternative Keyboards

Alternative keyboards permit users with physical disabilities to enter information into the computer with a keyboard that molds to individual needs. Basic adaptive adaptations that assist physically disabled students to use computers include replacing standard keys with larger keys that are easier to see and touch, reducing the number of keys on the keyboard, placing letters keys in alphabetical order, and providing keys that are brightly colored and easy to read. Other keyboards are much smaller than their traditional counterparts and have keyboard surfaces that are much more sensitive to touch.

These keyboards are excellent for individuals with a limited range of motion or for individuals who have a difficult time applying pressure to keys. Also, many alternative keyboards have a sticky-key feature to lock and hold Shift, Alt, and Ctrl keys, transforming keystrokes that ordinarily take two hands into a single-handed operation.

It is significantly to mention LOMAK a light operated mouse and keyboard. LOMAK will help empower physically disabled people who, with existing technology, can only input at speeds of between two and eight words per minute. With LOMAK they are able to achieve normal typing speeds. It is a very intuitive design making LOMAK easy and fast to learn.

LOMAK connects to an ordinary desktop or laptop computer without the need for additional software. It receives signals from a hand-operated pointer or a low powered laser pointer attached to a headband or cap worn by the user. He or she simply points at a letter on the keyboard and then to a central confirm key to input a keystroke.

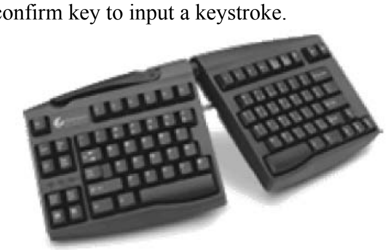


Fig. 1 GoldTouch USB Adjustable Ergonomic Keyboard

Touch Screens

Touch screens are very popular with young computer users and with individuals who have severe developmental or physical disabilities. The user needs only to point with a finger to make a selection. Many touch screens come complete with multiple screen overlays that can be used to perform a variety of tasks. Similarly, many companies provide additional software that enables the users to create their own overlays.

Fig. 2 IBM Touch Screen Monitor

High-precision touch screens increase the range of possible applications, especially if they are mounted in a position that is convenient for pointing and reading (30 to 45 degrees from the horizontal).

Infrared Sensors with Pneumatic Switches

Use of an infrared sensor worn on the head, along with use of a pneumatic switch, can enable physically disabled students to interact with the computer. As the user looks at the computer screen, the cursor follows the user's head movement. Moving the head to the left moves the cursor in the same direction on the screen. Thus, users can position the cursor anywhere on the screen by moving their head left, right, up, or down. This kind of switches are called head-pointer switches, because they are operated by relative head movements. The pneumatic switch, which is activated by inhaling or exhaling through a plastic tube, enables the user to use the mouse. When the user sips or puffs on the switch, the computer responds as if the mouse button had been clicked. In this manner, the user can move a cursor and click on items displayed on the computer screen. This kind of switches are called breath switches because they are operated by breath or sip-and-puff control. Special software is used in conjuction with these movements to allow the user to type out information on facsimile of a keyboard that is displayed on the computer monitor.

Voice Recognition

Using voice recognition software, the user can bypass the keyboard and just speak to the computer. By programming the computer with a set of predefined instructions, the user can control the computer by verbally issuing commands into a microphone. In most cases, the reliability of the system can be enhanced by having the user "train" the computer to recognize his or her speech patterns. Such systems are called speaker-dependent systems. Also, there are speaker-independent systems that are beginning to be reliable enough for certain commercial applications. Quiet environments, head-mounted microphones, and careful choice of vocabularies improve recognition rate. Voice recognition systems enable students to operate a variety of application programs, to dictate to a word processor, and to enter data into spreadsheets.

Recently, custom computer chips are used to improve speech recognition speed and lower its power consumption, as software has failed to overcome these problems. Faster chip-based speech recognition will enable video players to search rapidly for spoken words. And lower power consumption will enable a cell phone to take dictated notes. The speech recognition chip converts an audio signal into combinations of noises that form any of about 50 different sounds, like "n", in English. This is difficult as there are more than 1,000 sound possibilities. The chip then compares thouse sounds to those used in actual words. Lastly, the chip looks for likely combinations of words – both pairs and threesomes – for better accuracy. The chip performance is reliant on high memory communications bandwidth and hence it can make comparisons quickly.

Reading Systems

An individual who has a difficult time holding printed material or turning pages may benefit from a reading system. These systems are typically made up of hardware (scanner, computer, monitor, and sound card), Optical Character Recognition (OCR) software, and a reading/filling program. The system provides an alternative to reading printed text. Hard copy text is placed on the scanner where it is converted into a digital image. The image is then converted to a text file, making the characters recognizable by the computer.

The computer can read the words back using a speech synthesizer and simultaneously present the words on screen. Use of such a system may require assistance, since a disability that limits manipulation of a book may also preclude independent use of a scanner.

Fig. 3 VERA Reading System

Desktop Dictation

Desktop dictation, a technology that eases the hazards of typing in a computer by dictation from the desk, can be a long-term solution for the physically disabled.

As using speech is faster and easier than typing, Desktop dictation not only liberates you from the risk of developing a repetitive stress injury, but also gives the advantage of getting done almost everything routine in a computer such as drafting e-mail, letters, and memos.

Varieties of Desktop dictation products are available in the market today. In a world of survival of the best, the consumer selects a product considering the following factors: ease of use, compliance, accessibility, productivity, error frequency, ease of memorizing, and satisfaction.

For example, approaching the computer using minimum number of instructions and effort with Desktop dictation to do a function, accounts for accessibility. One of the reasons why many dictators stop using speech recognition is that it takes too long to put on the headset. Swapping the headset that comes with the software for a hand-held microphone, may solve this issue.

For those users who constantly switch between the headset for the phone and the headset for the computer, converting the headset to an integrated unit that will accommodate both devices, removes the disruption and makes the system usable.

If a computer has a high-speed processor and unlimited swap space on the hard drive or RAM, dictation can be a time saving and simple experience. If after talking into the microphone one will have to wait for a few seconds to get the text inserted into the document, it is unproductive, and distracting.

Using a dual core processor may improve dictation response time, simply because of the enhanced processing capabilities. According to sources at Hewlett Packard, usage of a dual core processor showed 30 to 50 percent improvement in non-speech applications when compared to the performance of a single core processor. This was effective in computers including the HP dv1000t notebook. Even though a product like Nuance's Dragon NaturallySpeaking is not made to multi-task, the dual core processor can add the advantage of improved dictation response time.

V CONCLUSION

The specific need for adaptive technology is unique to the individual [7]. Trial and error may be required to find a set of appropriate tools and techniques. The person with a physical impairment should play a key role in determining her goals and needs when selecting her adaptive technology. Once basic tools and strategies are initially selected, she can test drive, discard, adapt, and/or refine. The end user of the technology should ultimately determine what works best. For example, switches can be activated by any part of the body, allowing students with physical disabilities to control many aspects of their environment independently – from using a toy or radio for their own entertainment, to communicating with their nondisabled peers in the classroom, to controlling a computer or other high-tech or AAC device. Today, switches can be used with a number of adaptive devices that enable students with severe physical disabilities to successfully operate a computer independently, including turning the power on and off, inserting and removing a disk or CD from a drive, copying files, accessing a modem, and using a keyboard. A number of alternative input devices can be connected to a standard computer to assist or replace the use of a traditional keyboard, which is often the greatest barrier to computer use for students with physical disabilities. Adaptive keyboards, infrared sensors, and voice recognition systems, all have proven to be highly effective in helping students with physical disabilities use computers to participate in many educational activities that would not be available to them through other means. These devices range in price from less than 100 $ for some switches to as much as 9000 $ for higher-end, voice-activated systems. Trackballs, external touchpads, handheld pointing devices, and head-controlled pointing systems (such as HeadMouse™ or HeadMaster™) are some of possible choices that may be effective to control a cursor on the computer screen. Holography would present the important role in improving life's quality for persons with physical impairment.

So, people with physical impairments will be able to easily communicate with relatives and friends and be less lonely with face-to-face contact. It will be make their lives happier and easier. Further, Desktop dictation can be a long-term solution for the disabled. Optimizing its utility and productivity may improve the user experience. It seems that LOMAK has enormous potential for people who are unable to use a conventional keyboard. It adds a unique option to the assistive technology market.

The previously mentioned technologies have grown increasingly sophisticated and are becoming more familiar in classroom settings, and still other technologies are being developed for use in the near future. For example, a number of research labs are examining the use of devices such as robotic arms, which can help individuals who are physically disabled accomplish such daily activities as eating, retrieving objects, turning pages in books and magazines, and even play cards. Although it may be years before these technologies become commonplace, some robotic devices are already in use, and more sophisticated devices are continually under development. In time, they too may be commonplace, and technologies that have yet to be envisioned for use by students with severe physical disabilities will be moving into the limelight.

REFERENCES

[1] Joseph J. Lazzaro. *Adapting PCs for Disabilities*. Addison-Wesley Publishing Company, 1996.

[2] John R. Gunderson. *American with Disabilities Act (ADA): Human-Computer Interaction for Persons with Disabilities*. Conference companion on Human factors in computing systems. ACM Press. April 1994.

[3] Wolfgang Lutz, Sergei Scherbov. *Will Population Ageing Necessarily Lead to an Increase in the Number of Persons with Disabilities? Alternative Scenarios for the European Union*.

[4] Glenda E. Moum. *Accommodation the Computing Access Needs of Persons with Disabilities*. Proceedings of the 19th annual ACM SIGUCCS conference on User services. ACM Press. September 1991.

[5] Ben Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Third Edition. Addison-Wesley Longman, Inc., 1998.

[6] Jenny Preece et al. *Human-Computer Interaction*. Wokingham, UK. Addison-Wesley, 1994.

[7] John M. Carroll. *Human-Computer Interaction in the New Millennium*. Addison Wesley. ACM Press, New York. July 2001.

Mining E-Mail Content for a Small Enterprise

Emmanuel Udoh
Department of Computer Science
Indiana University-Purdue University
2101 E. Coliseum Blvd, Fort Wayne, IN 46805 USA

Abstract **– Emails constitute a rich source of a company's information, replacing fax, letters, phone and memos as the dominant form of inter- and intra-business communication. An email system is now a place where task is received, managed and delegated in a company. For small companies, the backbone of modern business, sieving and analyzing tons of business emails consume much business time. There are abundant data mining techniques that can assist email analysis. This paper describes a web-based approach to parse and mine email logs from a POP3 server for content information. The email system affords users a computer-aided tool for decision making.**
can be associated with diseases with improved visualizations.

Keywords **- Business Email, Data Mining, Decision Support System, JSP, Tomcat, Oracle.**

I. INTRODUCTION

Computer-aided decision making tools are essential for modern day business managers in view of the continued proliferation of information in electronic format, especially email messages. Managers spend a big chunk of office hours daily sieving through emails, since emails are now the dominant form of inter- and intra-business communication. Information once transmitted by phone, letters, memos or fax is now sent via email. From a company wide perspective, there is value in mining email content. Content analysis affords firms means to manage such large datasets and efficiently serve customers [3,4]. This paper focuses on mining emails as a business decision making tool. In this vein, the author is working with a small local engineering firm to mine email datasets, such as product information requests, technical support queries and job applications.

Content analysis seeks patterns in textual data [1,5]. Emails are mostly textual messages, but increasingly HTML and XML messages are also available. Email messages are derived from two different applications running on a server machine [1]. The simple mail transfer protocol server (SMTP with listener commonly at port 25) handles all outgoing emails, whereas post office protocol server (POP with listener at port 110) or Internet mail access protocol server (IMAP with listener at port 143) handles incoming mails. For mail mining purposes, POP3 or IMAP servers (provided by JavaMail distribution) are of interest. POP is one of the most popular mechanisms for retrieval of email messages and therefore likely to be of interest to the largest number of readers [1].

Researchers agree that harnessing the large email datasets requires data mining technologies [1,8]. There are research activities in content analysis of email for different purposes, such as automated filing, spam detection, filtering, and personalization services [7]. There are also email behavior analyses for forensic and intelligence purposes [8]. In some of these works, security tends to be the dominant consideration, since emails can be used for legitimate (document distribution) or illegitimate purposes [6,7]. Some misuse cases are virus applications, spambot activity and security policy violations [8]. These misuse examples can also be automatically detected by data mining techniques. But this paper focuses on an email system as a decision support system for a small business, barring anomalous activities by fraudulent Internet users.

An email mining system is a decision support system in the broad sense. It may not have all the paraphernalia associated with online analytical processing (OLAP) systems, but it can help managers compile information from emails for decision making. Thus, the primary objective of this work is to develop an online demo email system that harvests emails from a POP3 server and mines the content with techniques like classification, clustering and ranking for decision making. These techniques are essential because

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 179–182.

users naturally like to deal with the most important messages first, which can be obtained easily through classification, clustering and ranking.

II. TECHNICAL APPROACH

The developed prototype email system has conceptually two components – the email extraction unit and the data mining component. A database is used as a conduit or plays an intermediation role in facilitating the interaction between both components. Database is essential because archiving strategy affects retrieving efficiency. To actualize this approach, a three-tier architecture consisting of a browser front-end, a Tomcat application server middleware and an Oracle database backend was implemented. This web-based system has JSP providing the glue for the efficient execution of the logic layers as well as the background processes.

The extraction unit is a JSP package deployment running on a Tomcat server that connects to the email POP3 server, parses the email content, and eventually stores the output in an Oracle database server. As part of the JSP package, a mail parser wrapper provides a common interface to the disparate formats of the email protocols, determines the content type (plain text, HTML or XML) and ultimately invokes the correct parser.

Furthermore, the JSP package contains a filtering engine that picks out specific HTML, XML or plain text elements from the emails according to the positions or content of these elements within the structure. The filtered data can be stored or output to another unit. In addition, the extraction unit functionally sorts the emails separately, converts them to strings and stores them also in a vector for easy manipulation. Ultimately, the extracted data can be accessed by the data mining unit for manipulation after authentication.

The data mining unit visualizes the stored data in the Oracle database to the user. Currently, it implements three data mining techniques, namely classification, clustering and ranking. The algorithms first classify the emails into major components, then group them to clusters and finally rank them. Each method extracts different types of information from the email dataset. Clustering makes explicit the relationship between the emails, while classification identifies the key topics of the emails. Interested readers are referred to [1,3,4] for elaborate explanations of these algorithms. Based on the analysis of the email, profiles can be generated by extracting the frequencies of certain terms. Clustering and filtering can be carried out on the basis of both repetitive occurrence and co-occurrence, thereby providing a coherent picture of the functional relationship among large and heterogeneous email dataset.

III. RESULTS

To mine emails from the developed system, a user has to be authenticated by username and password. The system verifies the entry with the Oracle server, then logs into the POP3 server and retrieves all e-mails in the user's inbox/archive. Figure 1 shows the login screen, while Figure 2 shows email messages retrieved from the POP3 server. These e-mails are sorted into individual e-mails, converted into strings and then stored in the database. As one of the system functionalities, a user can extract information from the retrieved data, by selecting a type from a dropdown list (currently implemented are: Personal, Business, Accounting, Production and Engineering). Each type has an associated list of keywords that resides in a table in the database. Upon selecting a type, a result page (Figure 3) appears showing the total number of e-mails available, how many are now classified, a listing of all the keywords assigned to the type, available clusters and the number of hits for each word.

Another functionality provided by the system is the automatic clustering or labeling of the main topics instead of the selection as described above. The system does that by grouping appropriate keywords and detecting the topics through the parsing and extraction of the email content for easy decision making. It also exploits the domain knowledge and indexing for the retrieval of the needed information from the email datasets.

However, this system does not analyze emails for viral or spam activities. It presupposes that the emails are free from virus, spam and other malicious components. There are several approaches to achieve that, using for instance antivirus scanners like McAfee, Norton or Malicious Email Tracking (MET) systems. Many firms regularly use firewalls and antivirus scanners to monitor their email traffic. The developed prototype system can be used in conjunction with these antivirus programs.

Fig. 1: Web-based entry point to the email mining system

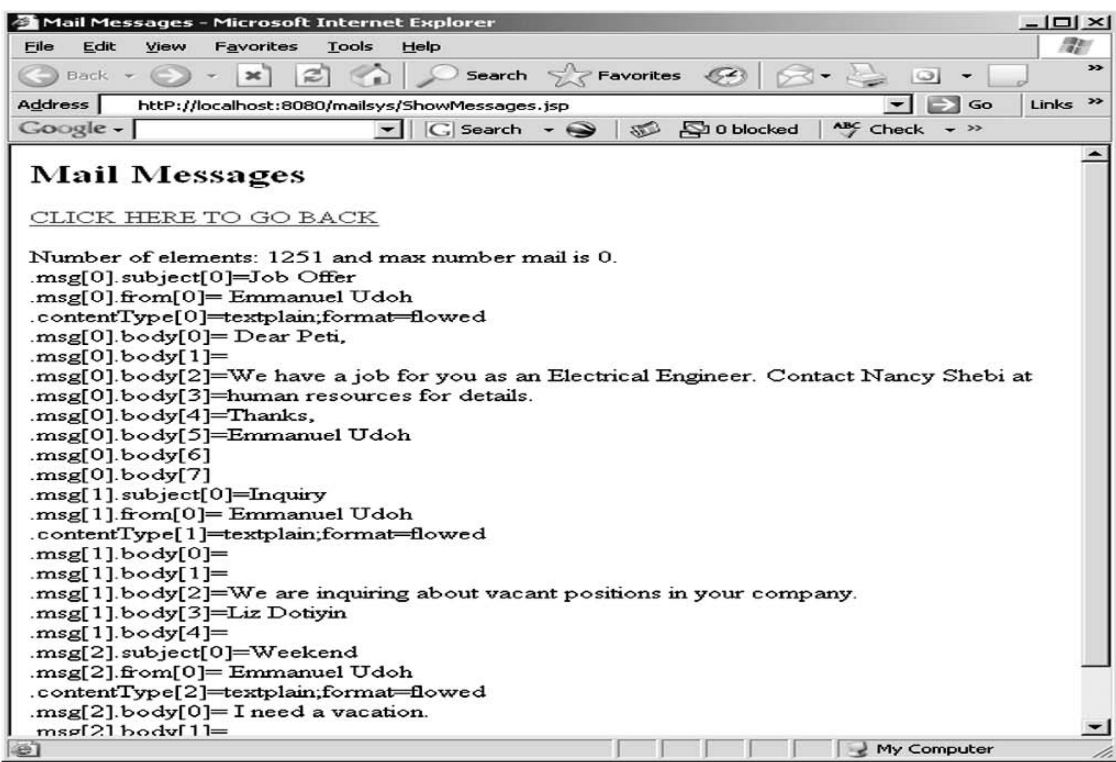
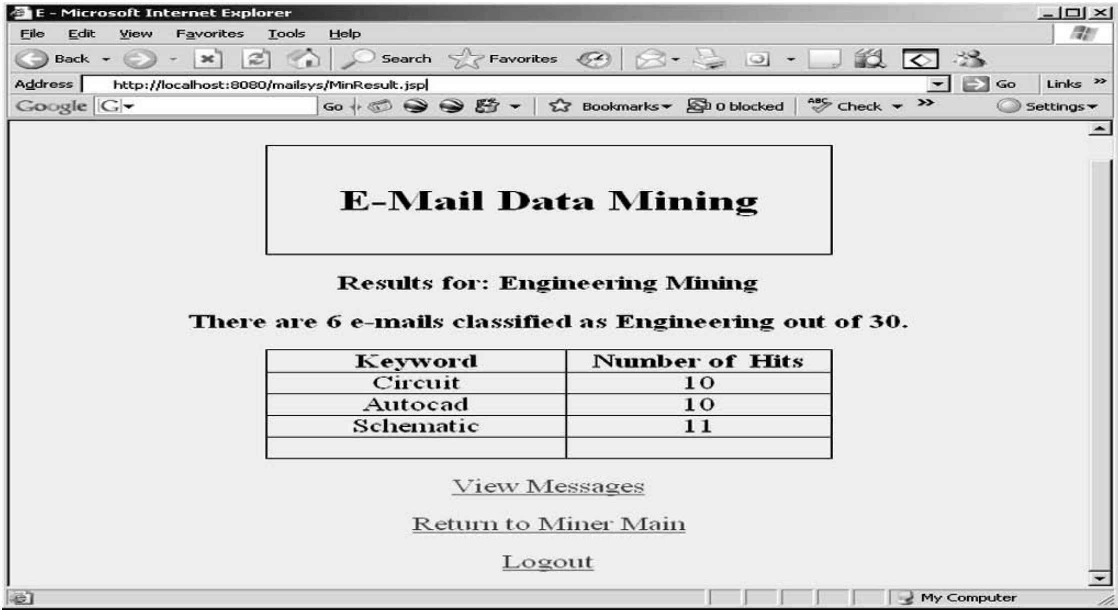


Fig. 2: Email messages retrieved from the POP3 server.



III. CONCLUSION

A prototype web-based email mining system was developed for the small business audience. It is a system that harvests emails from a POP3 server, classifies, ranks and clusters the content for decision purposes. The system does not examine the emails for misuse, and presupposes the emails are virus- and spam-free. It is an initial attempt to provide a simple functioning system that a small business manager can quickly use to synthesize emails that could have consumed precious business time. The system is still under development and will incorporate more robust statistical features in future versions

ACKNOWLEDGMENTS

This project was supported with a grant from the Rochester Medical Implant. For encouragement, the author thanks Dr. Erbach of the Department of Computer Science, Indiana University - Purdue University, Fort Wayne.

REFERENCES

[1] T. Loton, *Web content mining with Java*. (West Sussex, England: Wiley, 2002.)

[2] A. Scime, *Web Mining: Applications and Techniques*. (Hershey, PA: Idea Group Publishing, 2004).

[3] D. Sullivan, *Document Warehousing and Text Mining* (NY, NY: Wiley, 2003)

[4] A. Kao & S. Poteet, Text mining and natural language processing, *SIGKDD Explorations*, 7 (1): 2004, 1-2.

[5] R. J. Mooney & R. Bunescu, Mining knowledge from text using information extraction, *SIGKDD Explorations*, 7(1): 2004, 3-10.

[6] S. J. Stolfo, S. Hershkop, K. Wang, O. Nimeskern, and C. - W. Chia-Wei Hu, A Behavior-based Approach to Securing Email Systems. *Mathematical Methods, Models and Architectures for Computer Networks Security"*, Proceedings Published by Springer Verlag, 2003: 1-26., St. Petersburg, Russia.

[7] S. J. Stolfo, S. Hershkop, K. Wang, O. Nimeskern, and C. –W.Chia-Wei Hu, Behavior Profiling of E mail. *1st NSF/NIJ Symposium on Intelligence & Security Informatics,* Tucson,Arizona,USA. 2003:1-17.

[8] S. J. Stolfo and S. Hershkop, Email mining toolkit supporting law enforcement forensic analyses. ACM National Conference on Digital Government Research, 2005: 221 - 222 pp, Atlanta, Georgia.

QWERTY: A System of Logic and Symmetry?

Ernest Hibbs
mailto: emhibbs1@cox.net

***Abstract*-We may be able to finally say that we have cracked the code to the QWERTY keyboard design by finding both a logical and symmetrical mapping of the alphabet to the keys. We distinguish letters in the alphabet that have a right boundary characteristic. We use these letters to structure a pair of matrices whose column-selected sets directly map to a collage of symmetric QWERTY keyboard patterns. These matrix-selected sets can be used for teaching the QWERTY, for combining QWERTY sets with different soft keyboards, and possibly for assisting research related to cognitive informatics.**

I. INTRODUCTION

How could we possibly think of a QWERTY keyboard layout as having logic and symmetry? Believe it or not, what led us down this path was trying to find some visual characteristics of letters that could help teach our daughter the alphabet. An obvious pattern in the alphabet unfolded involving letters with a right boundary characteristic. We split the alphabet into two sub-matrices, with the stride between letters with a Right Boundary Demarcation Letter (RBDL) increasing from three to four in the second sub-matrix. The set of RBDL in the column of the first sub-matrix directly correlated to every other key in the QWERTY home row up to the letter 'k." We then forced the RBDL as a vector into the diagonals of the two sub-matrices to see what properties accompany a RBDL-focused alphabet matrix. As a result, we obtain twelve equally-spaced QWERTY patterns.

Although the teaching of QWERTY's history contains the technical issue of clearing the jams in the mechanical typewriter [1], we lack knowledge of how the letters were selected to be dispersed over our legacy QWERTY keyboard structure. Now with a connection to a defined RBDL matrix structure, teaching these matrix-selected sets can make the QWERTY keyboard easier to learn, make it contribute to combined keypad development, and make it a useful tool for further cognitive research.

II. INITIAL RBDL-QWERTY HOME ROW CONNECTION

We need to take an almost artistic view of the letters to initially determine the set of right boundary demarcation letters (RBDL). This is not a new concept. For centuries, communicating and understanding cultures has been done by using a written language's visual, symbolic, and artistic forms [2]. If we look at a vertical line as the boldest demarcation for ending a thought and add an arrow or flag from our English direction of reading (left), we now have the basic RBDL visual property. We are looking for an arrow (or flag waving) on the left-most body of a pole that indicates the point to either start or to stop. The letter "d" is a good example.

a	b	c		
d	e	f		
g	h	i		
j	k	l		
m	n	o	p	q
	r	s	t	u
	v	w	x	y
	z			

Fig. 1. Resultant RBDL Perpendicular Matrix Combination

The steps that initially connect us to the QWERTY are simple. We use the lower-case handwritten English Alphabet. This paper uses courier font in matrices to best demonstrate this visual property. Next, we traverse the alphabet and select the set of RBDL; we include the letter "m" only because it marks the middle. Our set of RBDL = {a, d, g, j, m, q, u, y}.

The result of using the set of RBDL to determine our matrix structure is a set of perpendicular matrices (Fig. 1). This matrix structure starts with a stride of three and increases the stride to 4 for the second sub-matrix (after the letter "m"). In Fig. 1, we have the 4x3 matrix and the 3x4 matrix in white boxes with its RBDL in grey-shaded boxes.

The first column of the first sub-matrix gives us our first connection to the QWERTY keyboard. The set {a, d, g, j} is also on the home row of the keyboard with a stride of two between the letters (Fig. 2). This seems too concrete to only be coincidental. We also see other sets from other columns that also provide some insight on the possible selection and placement of letters on the QWERTY.

In the three other selected sets for Fig. 2 ({x, p}, {o, w}, and {m, q}), we start to see the QWERTY keyboard as containing balanced, or symmetrical sets mostly selected from columns (Fig. 2). If we can extract patterns from a hard-wired legacy keyboard design (QWERTY), we can use those patterns in studies involving soft keyboard designs or as a tool to transition between the two designs. Current soft keyboard designs focus on building keyboards from the "[m]ost commonly used sequences" like "THE" and "ING" [3].

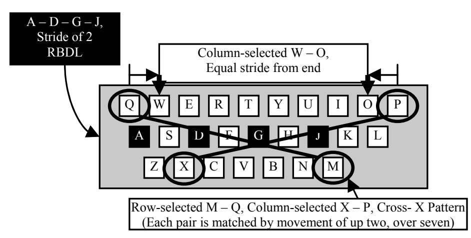


Fig. 2. Initial Perpendicular Matrix Symmetric Sets

T. Sobh (ed.), *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, 183–186.

In the second sub-matrix, when we select two pairs from the outer rows of the same columns ({x, p}, {o, w}), we see items from columns pushed to opposite ends of the keyboard. When we look at the set {x, p} along with the row relationship of set {m, q}, we see a cross pattern that "almost" fits perfectly. We could say this is a pattern of the pairs, where the connection between the pair goes up two rows on the keyboard and left (or right) seven strides. We get pattern-matching, but the location of "x" makes the overall QWERTY pattern look like an improperly buttoned shirt.

One of our goals should be to find symmetric patterns that can also be studied and compared to new keyboard patterns for performance of short term memory and long term memory, along with "[s]kill identification" [4]. Our hope is that the fully-deployed QWERTY may also become a greater tool for cognitive sciences. Is there a way to structure the RBDL-driven matrix to maximize QWERTY keyboard pattern matching? Yes. We will see that accomplished by using RBDL to force diagonal vectors as much as possible.

III. THE DRAFT QWERTY RBDL DIAGONAL MATRIX

If we were dealing with a numerical matrix, we could attempt typical matrix functions: such as diagonal, identity, eigenvectors, complements, determinants, or products [5]. Unfortunately, we have letters, all unique letters at that. So, let's build a matrix by forcing the set of RBDL on the diagonal to see if there are intrinsic QWERTY patterns communicated. Fig. 3, Fig. 4, and Fig 5 show the result.

Now things really get exiting! In our draft QWERTY RBDL diagonal matrix, there are twelve sets of characters that produce symmetrically distributed patterns on the QWERTY keyboard. We will see that the "ubiquitous" standard of the QWERTY is not a "frozen accident," but a frozen symmetrical design that we can reuse (references [6] and [7]). Let's use it to help transition to other soft keyboard approaches where "[t]he design is nearly symmetrical, making it suitable for either hand" [8].

We have the following three categories of patterns for ease of traceability, because displaying and overlaying all twelve patterns would be overwhelming: First Diagonal Sub-Matrix Symmetric Sets; Second Diagonal Sub-Matrix Symmetric Column-Selected Sets; Symmetric Structure and Perimeter Sets.

a	b	c	d
e	f	g	h
i	j	k	l
m	n	o	p
	q	r	s
	t	u	v
	w	x	y
	z		

Fig. 3. Draft QWERTY RBDL Diagonal Matrix

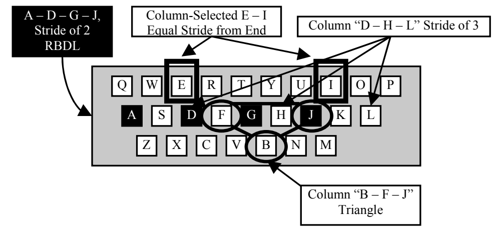


Fig. 4. First Diagonal Sub-Matrix Symmetric Sets

A. First Diagonal Sub-Matrix Symmetric Sets

Fig. 4 contains four sets of selected from the first sub-matrix of Fig. 3. We have already addressed the initial RBDL set {a, d, g, j}. We also need to note that two of these sets are not only symmetric sets on the keyboard, but also are entire column selections.

This indicates both a planned selection and a planned distribution. The {d, h, l} column-set having a keyboard stride of three allows it to start on "d" and also allows it to overlay the initial RBDL set. Why did column-set {e, i} not include the letter "a"? I have no idea. The letter "a" seems to be in a class all its own as a starting point both in the matrix and on the keyboard home row.

B. Second Diagonal Sub-Matrix Symmetric Column-Selected Sets

Now we will focus on the 3x4 sub-matrix covering the letters from "n" to "y." Fig. 5 has the QWERTY mapping for the second half of the matrix in Fig. 3. We select three pairs of letters and one set of two pairs of letters. We select from row one and from row the set of pairs {{n, w}, {o, x}} (we will visit {p, y} later), which forms the balanced "X" pattern in Fig. 5. We could look at the in-order pairs of {n, o} and {w, x} and also notice that these are one stride from the end. However, selecting in-order sets has not helped produce symmetric sets in the past – otherwise we already could read obvious and complete in-order patterns from our QWERTY keyboards.

The next pair is in the center column on the two middle rows. The pair {r, u} is symmetrically selected and found symmetrically placed in the top row of the QWERTY keyboard. We also have a set of two pairs {{q, t}, {p, y}} that

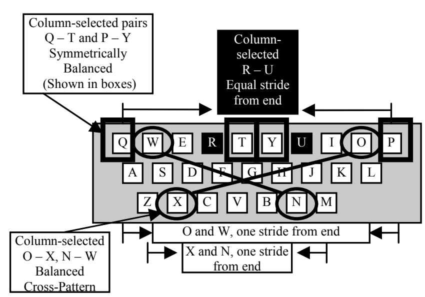


Fig. 5. Second Diagonal Sub-Matrix Symmetric Column-Selected Sets

are equally spaced between their letters and whose pairs are symmetrically distributed on the top row of the keyboard. Now with all these symmetric patterns, what else could we possibly find? The answer: sets that actually map to the perimeter and structure of the two sub-matrices.

C. Symmetric Structure and Perimeter Sets.

If we were describing the structure for each of these alphabet sub-matrices, we would need to label the beginning, the middle, and the end for each sub-matrix. Fig. 6 does just that. The set containing the beginning, middle, and end of the first sub-matrix is {a, g, l}, which is perfectly spaced on the middle row of the keyboard at a stride of four.

The second sub-matrix has two structural patterns on the keyboard. It has a symmetric "m-t-z" triangle for its beginning, middle, and end. It also a separate set for three perimeter corner letters ("p," "y," and "w") of the 4x3 section in the second sub-matrix. Whether coincidental or planned, these three corners are found on the top row of the keyboard equally spaced at a stride of four. There is also "p-q-v" triangle in the matrix that is also on the QWERTY keyboard.

Probably some of these symmetric structure and perimeter relationships are merely coincidental. Even so, the overall selection approach still seems very well-organized. Now, let's go on to optimize the matrix by addressing the three unmapped letters.

D. Final RBDL Diagonal and QWERTY Matrix

When we look at the location of our three letters ("c," "k," and "s") remaining to be mapped to a pattern, we immediately notice the set {s, k} symmetrical in the home row. Now what? Now we revisit our draft QWERTY RBDL diagonal matrix (Fig. 3) to do two things: maintain all the found symmetrical relationships and find a new set of relationships that focuses on the combination producing the set {s, k}.

Can this be done? Yes, we need to first pivot the second sub-matrix 90-degrees clockwise. This keeps the same matrix dimensions, but now we traverse down, or column-wise, after the middle letter "m." Next, in Fig. 7, we look at the second sub-matrix in a different manner. Imagine we pick it up and place it on top of the first sub-matrix, where "n" is over "d" and "w" is over "a." We just achieved two things: "s" is over "k" and we have an almost perfect diagonal of all the RBDL, except for the letter "a."

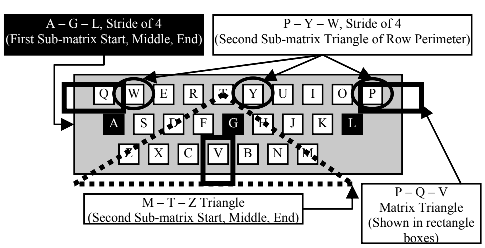


Fig. 6. Symmetric Structure and Perimeter Sets

We still have the letter "c" unmapped, but this is probably about as close of as we can get to a totally coordinated QWERTY design matrix. Of course, our column-sets in the second sub-matrix are now row-selected sets. Even those differences are acceptable, since the symmetric relationships still exist, just in a row-wise form. But now, with a column-wise traversal in the second matrix, we have returned full-circle to an underlying theme of perpendicular relationships between the two sub-matrices.

IV. THE PERPENDICULAR THEME OF THE MATRICES

We notice that with every variation of the RBDL-driven matrices there is a perpendicular theme. In Fig. 1, the 4x3 portion of the first sub-matrix is perpendicular to the 3x4 portion of the second sub-matrix. In Fig. 3, not only are the two sub-matrices perpendicular, but the RBDL diagonals of the two sub-matrices are also perpendicular. In Fig. 7, where the diagonals are aligned, two perpendicular relationships still exist: the sub-matrices' traversals and the selection of QWERTY sets. Are there any other possible reasons for this reoccurring perpendicular theme? Are perpendicular information structures more cognitively palatable? I don't know. We may want to consider further research on the topic. The next two paragraphs give an example of the possible type of cognitive research content.

Fig. 1 may show a rudimentary cognitive "handling" of a language. By using the term "handling," we refer to the actual cognitive processing of items as if placed in with our left and right hands. Left-brain and right-brain cognitive properties are really localized processors of certain functions. However, in any one function, mathematics for example, we process one-sided brain function with both hands. Okay…so what does this have to do with Fig. 1?

Looking at Fig. 1, we can imagine placing our left 4 fingers over the rows beginning with "a," "d," "g," and "j." Next, we look at the second sub-matrix and imagine placing our 4 right fingers pointing up the columns with the letters "v," "w," "x," and "y." The column with "q-u-y" is now under the small finger of our right hand. With a letter on each of the three sections of our two sets of four-fingers, we grab the alphabet. So, do we grab onto, or "handle," our other concepts in the same manner? I don't know. This type of left-hand and right-hand perpendicular structure may be of interest to professionals in the Cognitive Sciences or Cognitive Informatics.

				a	b	c	d
				e	f	g	h
				i	j	k	l
				m			
z	w	t	q	n			
	x	u	r	o			
	y	v	s	p			

Fig. 7. Final Optimal QWERTY RBDL Diagonal Matrix

V. THE NEXT STEPS

Now that our almost archeological dig has led to the discovery of the QWERTY keyboard's RBDL diagonal matrix, how can this matrix help us today? We have the advantage over people in the last century of knowing the specific directions that technology has greatly advanced in the 130 years since the typewriter's first public use. One of the more significant advancements is in the form of quality management techniques. We now have this new-found structure to assist in management, in technology transition, and in research in sciences for the common good of our society.

A. A Good Management Practice

This RBDL structure can help us take us from a Machiavellian approach of task definition to a quality management approach of empowering the workforce with more knowledge. In the gallows of the workforce, the QWERTY has been the oar that we must move to the beat of our very short schedule-driven projects. A workforce educated on the design patterns of the QWERTY will find innovative ways to make those patterns feed productivity. We already have smart people eager for better performance.

B. A Good Transition Tool

What about the life cycle of the QWERTY? Will we replace it with a better keyboard? Maybe, but we need a smooth transition. The future keyboard designers can reuse the RBDL to as an intermediate step away from the QWERTY, just as sets of letters are now mapped to a numeric keypad on our phones. We can use the RBDL to move us to a better keyboard that is logically built for input performance, similar to the approach taken in Japan [9].

C. A Tool for the Common Good

Are there other areas in Cognitive Science that can also benefit with either this right-boundary feature of the alphabet or the selected symmetric RBDL sets? If these RBDL actually carry more weight that the rest of the alphabet, should a new form of visual text communications arise? Maybe very strong concepts or safety steps would be best conveyed by using semantically equivalent words with a RBDL start, end, or both. Can we assist and guide learning disabilities with either the RBDL approach or the QWERTY symmetric sets? Can the profession of Cognitive Informatics use the RBDL QWERTY sets to guide the selected "menu" of "patterns" used in previous studies on soft keyboards [10]?

What about the perpendicular theme of the sub-matrices due to the different RBDL strides? Is there an associated left and right hand structure that we naturally build when ordering or learning? It makes us wonder.

Whether our individual professions and intentions are for the betterment of the workforce, the transition to new technology, or tools to assist research for the common-good, we all must look at the QWERTY keyboard differently. Only we can make it an applied system of logic and symmetry.

VI. CONCLUSION

What can we take from this paper that will benefit us? We have seen that there are physical characteristics in the lower-case English Alphabet that can form matrix patterns for teaching and for symmetrically distributing sets of letters. It appears that a diagonal RBDL matrix may very well have been used in the method to distribute the letters on the QWERTY keyboard. We can confidently say that the configuration of the QWERTY is not by random selected sets or random separation of letters. Ironically, QWERTY does contain a system of symmetry and logic.

Significant to the future and the reuse of the QWERTY is a RBDL pattern matrix that serves as a common logical structure of the QWERTY. At a minimum, we owe our fellow professionals a logical connection to symmetric patterns that will help them not feel oppressed without available knowledge of their daily tool. These newly discovered (or rediscovered) RBDL and column selected sets should be put to the test by serving both as tools for transitioning from the QWERTY and as tools for research by Cognitive Informatics. We may find it hard to ever look at the QWERTY keyboard as we have in the past. However we use it, the embedded logic and symmetry of the QWERTY keyboard is a system available for the advancement of knowledge in our professions.

REFERENCES

[1] S. Condoor, "Importance of teaching the history of technology," [*IEEE 34th Annual Frontiers in Education Conf. Savannah, GA*, p. T2G-9, 2004].

[2] J. Drucker, *The Alphabetic Labyrinth, the Letters in History and Imagination.* London: Thames and Hudson Ltd, 1995, pp.11-12, 93.

[3] S. MacKenzie, S. X. Zhang, "The design and evaluation of a high-performance soft keyboard," [*ACM SIGCHI Conf. on Human factors in computing systems Pittsburgh, Penn., USA*, p. 27, May 1999].

[4] A.Tsubokura, N.Ashida, N.Matsubara, I.Oshima, K.Kozuki, and K.Tsushima, "Analysis of key typing process using hand position process," [*IEEE Proceedings of the International Conf. on Computers in Education*, p. 687, 2002].

[5] D.C. Lay, *Linear Algebra and Its Applications*, 2nd ed. Reading, Massachusetts: Addison Wesley Longman, Inc., 1997, pp. 44, 191, 297, 313, 375, and 443.

[6] W. Smith, "Who dictates your display terminal design?" *ACM SIGCHI New York, NY, USA,* bulletin 16, issue 1, p. 6, July 1984.

[7] M. Milne, "Beware of the QWERTY," *ACM SIGGRAPH Computer Graphics USA*, vol. 32, issue 1, pp. 8-10, February 1998.

[8] S. MacKenzie, S. X. Zhang, "The design and evaluation of a high-performance soft keyboard," [*ACM SIGCHI Conf. on Human factors in computing systems Pittsburgh, Penn., USA*, p. 27, May 1999].

[9] Y. Hiraga,, Y. Ono, Yamada-Hisao, "An assignment of key-codes for a Japanese character keyboard," Assoc. for Computational Linguistics, 8th Inter. Conf. on Computational Linguistics Tokyo, Japan, p. 249, 1980.

[10] P. Isokoski, "Performance of menu-augmented soft keyboards," *ACM SIGCHI Conf. on Human factors in computing systems Vienna, Austria,* vol.. 6, Num. 1, p. 427, April 2004.

Analyzing the Statistical Behavior of Smoothing Method

Feng-Long Huang[1], Ming-Shing Yu[2]

[1]Department of Computer Science and Information Engineering,
National United University,
MiaoLi 360, Taiwan. flhuang@nuu.edu.tw

[2] Department of Information Science, National Chung-Hsing University,
Taichung 40227, Taiwan. msyu@nchu.edu.tw,

ABSTRACT

In the paper, we address the issues for *Good-Turing* smoothing method. Five properties are proposed and employed to analyze the statistical behaviors of *Good-Turing* smoothing method. Because of the violation of the *Good-Turing* smoothing, the related problems should be resolved to keep the method can be used normally.

The problem of zero n_c can be resolved by setting *cut-off k*. The number of n_c and recount c^* of English words and Mandarin character bigrams model are also listed. Three models, *character unigram, bigram and word unigram* model are generated for zero number of n_c .and *bigram* model is used to compare the entropty *H* based on various *cut-off k*.

I. Introduction

The model for word sequence prediction is the n-gram. *n*-gram model uses previous *n*-1 words to predict the next word. In speech recognition, it is always represented as the term language model (LM), [1],[3],[13].

Language models have widely been used in various tasks of natural language processing (NLP), such as speech recognition, machine translation, part-of-speech tagging, spelling correction and word sense disambiguation, etc, [2],[5],[10],[16]. Many research works of language modeling approach, such as [23], [24] and [25], have been used on information retrieval (IR). An event can be regarded as a possible type of *n*-gram in LM, *n*=1, 2, 3. We can calculate the probability for the each occurred event according to its count in training corpora. Because there may be unseen events in testing word sequence, smoothing method should be usually needed to re-estimate prior probability for each event.

A. *Language Models*

For a source sequence *Str*, the most probable target sequence *W* will be predicted, denoted $w_1,w_2,w_3,\ldots,w_m$ or w_1^m , where *m* is the number of words. Many different word sequences are possible for sequence *Str*.

A language model will be used to decide the correct target word sequence *W*. The prior probability P(*W*), where $W=w_1w_2w\ldots w_m$ is a possible translation of *Str*, can be represented as:

$$P(w_1^m)=P(w_1^1)P(w_2\mid w_1^1)P(w_3\mid w_1^2)..P(w_m\mid w_1^{m-1})$$

$$=P(w_1^1)\prod_{i=2}^{m}P(w_i\mid w_1^{i-1}). \tag{1}$$

The prior probability for a given word w_i in Eq. (1) are computed based on a long sequence of preceding words w_1^{m-1} . It is apparent parameter space is so large that a big corpus is needed. One way to resolve this problem is to estimate the approximate probability of a given word by using the $(n\text{-}1)^{\text{th}}$ preceding word sequence. For example, the model with *n*=1 (unigram) can be expressed as:

$$P(w_1^m)=\prod_{i=1}^{m}P(w_i) \tag{2}$$

The probability model with *n*=2 (bigram) can be expressed as:

$$P(w_1^m)=P(w_1)\prod_{i=2}^{m}P(w_i\mid w_{i-1}). \tag{3}$$

As shown in Eq. (3), the probability for each event can be obtained by training bigram model (*n*=2). Therefore the probability of a word bigram *b* will be written as:

$$P(w_i\mid w_{i-1})=\frac{C(w_{i-1}w_i)}{\sum_w C(w_{i-1}w)} \tag{4}$$

where $C(w_i)$ denotes the count of word w_i in training corpus. The probability $P(\cdot)$ of Eq. (4) is the relative frequency and such a method of parameter estimation is called *maximum likelihood estimation* (MLE).

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 187–192.

B. Unseen Event: Zero count issue

According to definition of the Eqs (2) and (3), the word sequence *W* can estimated based on MLE. However, such method may lead to the degradation of performance because of unseen events.

For a given word in bigram model, if the so-called the unseen event, which don't occur in training corpora, appear, then $C(w_{i-1}w_i)$ of such event is equal to 0 and Eq. (4) is also equal to 0. It is obvious that the P(*W*) in Eqs (2) and (3) are 0.

C. Smoothing Motheds for LMs

The unseen events always exist in NLP. Basically, the number of unseen events decreases while size of corpora increases. It is not reasonable to assign 0 to the unseen events. If we should assign certain probability to such unseen events, how the probability is assigned? The probability obtained from MLE should be adjusted and redistributed. Such a process must make the total probability for all known and unseen events to be unity. The schemes used to resolve the problems are called *smoothing methods*; which are usually employed to alleviate the zero-count issue in LMs.

There are many well-known smoothing methods, such as Additive discounting, Good-Turing, Witten-Bell, Katz[1] and so on, [4],[7],[9],[11],[12],[15],[18],[22]. In the paper, we will focus on Good-Turing smoothing and analyze its statistical features for three Mandarin LMs.

II. The Properties for Analyzing Statistical Behaviors of Language Models

In this section, we propose five properties which can be regarded as statistical features of LMs. These properties will be further used to analyze the statistical behaviors of smoothing methods in next section.

1. Property 1

The smoothed probability for any one bigram b_i should falls between 0 and 1 (0,1), as follows:

$$0 < P^*_{i,N} < 1 \quad \text{,for all bigrams } b_i \text{ on any } N \tag{5}$$

where $P^*_{i,N}$ is the smoothed probability for a bigram b_i (or word w_i) on training size *N*, *B* is the total number of types of bigrams.

2. Property 2

The summation of smoothed probability P^* for all the bigrams is necessarily equal to 1 on any training size *N*. Total smoothed probability *P* is summed as:

$$P^*_{1,N} + P^*_{2,N} + \ldots + P^*_{B,N} = \sum_{b_i \in seen\ bigrams} P^*_{i,N} + \sum_{b_j \in unseen\ bigrams} P^*_{j,N} = 1, \tag{6}$$

where *B* denotes the total number of bigrams.

3. Property 3

The smoothed probability assigned to the bigrams *b* with different count should satisfy all the following inequality equations[2]:

$$Q^*_{c,N} < Q^*_{c+1,N}, \quad \text{For } c=0,1,2,\ldots, \tag{7}$$

where $Q^*_{c,N}$ is the smoothed probability for the bigram b_i with *c* counts on training corpus of size *N*.

Inequality Eq. (7) describes the concept that smoothed probability for any bigram with same count should be same on any training size *N*. Furthermore, the probability for bigram b_{c+1} with *c*+1 counts should be larger than that of bigrams with *c* counts.

4. Property 4

Comparing to the probability *P* prior to smoothing process, the smoothed probability P^* for all bigrams will be changed. Property 4 can be expressed as follows:

$$Q^*_{0,N} > Q_{0,N}, \quad \text{for } c = 0 \tag{8}$$

$$Q^*_{c,N} < Q_{c,N}, \quad \text{for } c \ge 1 \tag{9}$$

Property 4 shows $Q^*_{0,N}$ for unseen bigrams will be larger than original while $Q^*_{c,N}$ will be decreased for all bigrams with more than one count (*c*>=1). The probability mass P_{mass} discounted from all known bigrams is distributed uniformly to the smoothed probability for unseen bigrams.

5. Property 5

Three notations *B*, *S* and U can be expressed as *B*=*S*+*U* for bigram models. When the number of training size is increased, all the smoothed probability Q^* for bigrams with same counts on training size *N*+1 should be decreased a bit while comparing to the Q^* on training size *N*. For instance, when an incoming bigram (say b_{N+1}) occurs, the training size is increase by one (now *N*=*N*+1). The smoothed probability Q^* on *N*+1 training set should be less than the probability Q^* on *N* for $c \ge 0$, except the P^* for the incoming bigram b_{N+1}:

$$Q^*_{c+1,N+1} = \frac{c(\bullet)+1}{N+1}\ . \tag{10}$$

In other words, in addition to the Q^* of b_{N+1} at training size *N*+1, all other smoothed probability Q^* at training size *N*+1 will be decreased than those at training size *N*. Although both the numerator and denominator of Eq. (10) are increased by 1, due to *N*>>*c*, so the inequality equation $Q^*_{c,N} < Q^*_{c+1,N+1}$ will hold. In summary, property 5 can be expressed as:

$$Q^*_{c,N} > Q^*_{c,N+1} \quad \text{for all bigrams with count } c >= 0 \tag{11}$$

$$Q^*_{c,N} < Q^*_{c+1,N+1} \quad \text{for the new bigram } b_{N+1}. \tag{12}$$

[1] In the paper, we employ the *Good-Turing* smoothing method to discount the count *c* for all events.

[2] The property was first proposed in [17] and we make a little modification.

where $Q^*_{c,N}$, $Q^*_{c+1,N+1}$ denote the smoothed probability for bigram with c counts on training size N and N+1. Note that $Q^*_{0,N}$ denotes the smoothed probability for unseen bigrams with 0 count.

III. Statistical Properties Analysis for Good-Turing Smoothing

A. Smoothing Processes

In general, the principal idea of smoothing is to adjust the total probability of seen events to that of unseen events, leaving some probability mass (so-called escape probability, P_{esc}), for unseen events. Smoothing algorithms can be considered as discounting some counts of seen events in order to obtain the escape probability P_{esc} which will be assigned to unseen events. Basically, the adjustment of smoothed probability for all occurred events involves discounting and redistributing processes:

1). Discounting:

The probability for all seen and unseen events is summed to be 1 (unity). First operation of smoothing method is the discounting process, which discount the probability of all seen events.

2). Redistributing:

In this operation of smoothing process, the escape probability discounted from all seen events will be redistributed to unseen events. The escape probability is usually shared by all the unseen events. That is, the escape probability is redistributed uniformly to each unseen event, P_{ESC}/U, where U is the number of unseen events. On the other hand, each unseen event obtains same probability.

B. Good-Turing Smoothing Method

Good-Turing smoothing is first described by Good in 1953 [7]. Some previous works are [3], [8] and [14]. Notation n_c denotes the number of n-grams with exactly c count in the corpus. For example, n_0 represent that the number of n-grams with zero count and n_1 means the number of n-grams which exactly occur once in the corpus of size N. Therefore, n_c will be described as:

$$n_c = \sum_{w:C(w)=c} 1 \tag{13}$$

where w denotes a bigram in corpus.

Based on the *Good-Turing* smoothing[7], redistributed recount c^* (so-called the smoothed count) will be presented in term of n_c, n_{c+1} and c as:

$$c^* = (c+1)\frac{n_{c+1}}{n_c} \tag{14}$$

Note that the numerator in Eq. (4) will be substituted by c^* for the bigram models as:

$$P(w_i \mid w_{i-1}) = \frac{c^*}{\sum_w C(w_{i-1}w)} \tag{15}$$

As we discuss in Section 1, the zero count of bigram will occur. For a given word in bigram model, the unseen event can lead to zero count issue.

B. Statistical Properties Analysis

Based on the 5 proposed properties above, we will analyze statistical behavior for Good-Turing Smoothing in the subsection.

Total number N of count in training corpora can be computed as:

$$\sum_i c_i n_i = c_0 n_0 + c_1 n_1 + c_2 n_2 + c_3 n_3 + \ldots = N, \text{ for all } i >= 0.$$

Property 1, 2 and 3 does not holds. For instance, in the following case: when $n_m = 0$,

$$c^*_{m-1} = (m-1+1)\frac{n_m}{n_{m-1}} = 0$$

$$c^*_m = (m+1)\frac{n_{m+1}}{n_m} = \infty$$

The two formulas above violate the propertyt 1 and 2, referred as Eqs 5 and 6. In such a situation, it is obvious that $Q^*_{m-2,N} > P^*_{m-1,N}$ and $Q^*_{m,N} > Q^*_{m+1,N}$. Hence, the results also violate the property 3.

It is possible that one of n_m for certain amount of training set is zero. The smoothed probability for unseen and known bigrams with c counts, property 4 does not hold.

When a new bigram b_{N+1} is read in, then training size is increased by one (N=N+1). As shown in Eq. (14), the smoothed recount $c^* = (c+1)\frac{n_{c+1}}{n_c}$. Supposed that the bigram b_{N+1} is ever known with c counts on training size N, upon the b_{N+1} appears, N=N+1, n_c=n_c-1 and n_{c+1}=n_{c+1}+1, the smoothed probability for bigrams with c on training size N and N+1 can be computed as:

$$Q^*_{c,N} = (c+1)\frac{n_{c+1}}{n_c}/N \quad \text{and}$$

$$\frac{Q^*_{c,N}}{Q^*_{c,N+1}} = \frac{(N+1)\frac{n_{c+1}}{n_c}}{N\frac{n_{c+1}+1}{n_c-1}} = \frac{(N+1)(n_c-1)n_{c+1}}{Nn_c(n_{c+1}+1)} \tag{16}$$

According to inequality Eq. (11), $Q^*_{c,N} > Q^*_{c,N+1}$. In other words, Eq. (16) should be greater than 1. In fact, N>>n_c and N>>n_{c+1}. Therefore, Eq. (16) may be < 1 on certain situation, although it is also possibly greater than 1. Hence, property 5 does not hold.

For the bigram b_{N+1}, what is the relation between the smoothed probabilities Q^* on training size N and N+1? We have:

$$Q^*_{c,N} = (c+1)\frac{n_{c+1}}{n_c}/N \quad \text{and} \quad Q^*_{c+1,N+1} = (c+2)\frac{n_{c+2}}{n_{c+1}+1}/(N+1)$$

then,

$$\frac{Q^*_{c,N}}{Q^*_{c+1,N+1}} = \frac{(c+1)(N+1)n_{c+1}(n_{c+1}+1)}{(c+2)Nn_c n_{c+2}} \tag{17}$$

According to Eq. (12) in property 5, Eq. (17) should be less than 1. It is obvious that Eq. (17) may be greater than 1 in certain situations, while it is possibly less than 1. Therefore, property 5 does not hold.

In addition to the Good-Turing smoothing, the properties for several other well-known smoothing methods have been further analyzed. Due to the paper size, they are omitted. Table 1 only shows the analyzed results for five well-known smoothing methods, in which O and X represent the property hold and not hold, respectively.

As a consequence, five property violations can be found for Good-Turing smoothing methods.

IV. Experiment Results

Good-Turing has been employed in many natural language applications. Previous works [3] and [13] discussed the related parameters, such as cut-off k, to avoid the zero n_c problem in *Good-Turing*. However, these works employ English corpus only. In this section, we will focus on the *Good-Turing* method in Mandarin corpus and further analyze the problems for Mandarin texts, such as cut-off k and recounts c^* for known and unseen bigrams.

A. Data Sets and Empirical Models

Two text sources are used as data sets; the news texts collected from Internet and ASBC corpus [18]. The HTML tags and all unnecessary symbols are extracted and there are about 7M Mandarin characters in news texts. The Academic Sinica Balanced Corpus version 3.0 (ASBC) includes 316 text files distributed in different fields, occupying 118MB memory and about 5.22 millions of words labeled with a POS tag. Our corpus contains totally up to 12M Mandarin characters.

Three models are constructed to evaluate the entropy of smoothing methods discussed in the paper; Mandarin character unigrams, character bigrams and word unigrams model. The entropy of each method is calculated on various data size in our experiments, from 1M to 12M Mandarin characters. The first two models employ up to 12M Mandarin characters (unigrams and bigrams) and the 3rd model use about up to 5M Mandarin words in ASBC corpus.

B. The First Emitting Zero n_c of Event Count

The number of n_c and recount c^* of English words and Mandarin characters (12M) bigram model are listed in Table 1. Church and Gale [4] used the 22M English corpus from Associated Press (AP) to calculate the recount c^* of character bigrams. As shown in Table 1, 4.49E+5 bigrams occur twice (count c=2) and the recount c^*=1.26. In the paper, we employee 12M Mandarin news texts collected from web site for bigrams by *Good-Turing* method. There are 1.34E+5 bigrams with a count c of 2 and its recount c^* =1.51. It is obvious that the equivalent recounts c^* for Mandarin characters are larger than that of English words through all the counts c.

Table 2: The events number n_c and recount c^* by *Good-Turing* discounting for English word bigrams and Mandarin character bigram models.

models / count c	English[3] Bigrams (22M)		Mandarin char. bigrams (12M)	
	n_c	c^*	n_c	c^*
0	7.46E+10	2.70E-5	1.69E+8	2.11E-3
1	2.01E+6	4.46E-1	3.57E+5	7.50E-1
2	**4.49E+5**	**1.26**	**1.34E+5**	**1.51**
3	1.88E+5	2.24	6.81E+4	2.58
4	1.05E+5	3.24	4.39E+4	3.54
5	6.83E+4	4.22	3.12E+4	4.47
6	4.81E+4	5.19	2.33E+4	5.51

In Table 2, the first zero of n_c and its count c for three Mandarin models are listed. The first emitting zero of n_c, for instance, will occur at count c=113 in training size N=1M (10^6) for Mandarin characters unigram model (*S1*). All the n_c with respect to count <113 will be a certain integer number. However, some n_c with respect to the count >113 may be zero; in such a situation, the case described in previous section will appear and. lead to the property violation.

In Table 2, *S1*, *S2* and *S3* denote the following models: char. unigram, char. bigram and word unigram model, respectively.

Table 2: The count c with respect to first emitting zero z for n_c with respect to various N in size of Mega (10^6)

N / model	1	2	3	4	5	6	7	8	9	10	11	12
S1	*113*	132	110	133	167	86	204	154	129	214	111	172
S2	226	269	325	356	433	457	471	498	634	592	568	590
S3	208	292	304	401	448							

C. Cut-off k and Entorpy

One problem for *Good-Turing* is zero value of n_c with count c for a bigram. Because the recount c^* is

[3] 22 million ($2.2*10^7$)words bigrams from Associated Press (AP).

calculated based on Eq. (14), if the denominator n_c is zero, the recount c^* will be infinitive (∞) and lead to infeasible smoothed probability P^*, as described in previous section. Infinitive probability P^* for any event is unacceptable and also violate the proposed property 1 and 2 (as discussed in this section).

Another problem; the negative recount c^*, will happen in some situations for *Good-Turing* smoothing method. Referring to empirical results, some recounts c^* are negative (<0). In such case, furthermore it leads to negative probability P and such statistical behavior also violates the proposed property. This situation also happens to some other recounts in character unigram model.

Therefore, the cut-off k should be defined to avoid its occurrence of wo problems discussed avobe. In other words, the type number n_c of bigrams with respect to count c < cut-off k shouldn't be zero and the recount c^* will be positive (larger than 0). In practice, not all the c^* are useful. In the empirical observation, large count c (for all c > cut-off k) are basically reliable and only all counts c less than k need to be recounted. Katz suggest that k=5 in English text corpus.

Figure 1 shows the Entropy H and various cut-off k for model *S2* with respect to the various training size N, from 1M to 12M Mandarin characters. In order to analyze the zero n_c of type number, we implement the experiment of first zero n_c for two models, in addition to character unigram model due to the occurrence of negative recount.

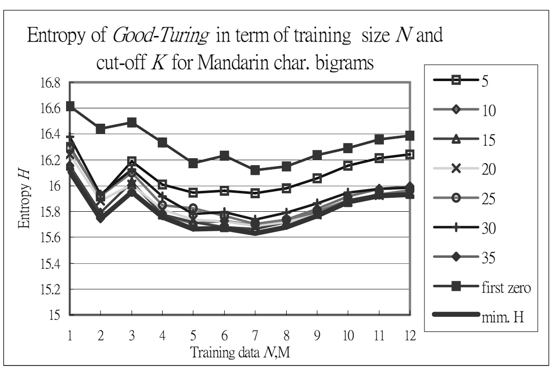


Figure 1: Entropy H and cut-off k in term of training size N for Mandarin character bigrams (*M2*).

For the character bigram model *S2*, if the cut-off k is set to first zero of n_c, the entropy H is higher that that of smaller cut-off value. For the word unigram model, if cut-off k is set to first zero of n_c, the entropy H is smallest among all different k.

V. Conclusions

In the paper, we address the issues for *Good-Turing* smoothing method. Five properties are proposed and employed to analyze the statistical behaviors of *Good-Turing*. Because of the violation of the smoothing method, the related problems should be resolved to avoid the degradation of performance.

The problem of zero n_c can be resolved by setting cut-off k. The number of n_c and recount c^* of English words and Mandarin character bigrams model are also listed. Three models, *S1*, *S2* and *S3*, character unigram, bigram and word unigram model, are generated for n_c. and *S2* is used to compare the entropty H based on various cut-off k.

In the future, we will compare the related parameters of *Good-Turing* to find better performance for Mandarin LMs, and then employ the smoothing method on several practical tasks, such as word segmentation, Part-of-speech (POS) tagging and word sense disambiguation (WSD).

ACKNOWLEDGEMENT

Authors thank Academic Sinica of R. O. C. for the corpus of ASBC, and Dr. Piyu Tsai for his comments.

REFERENCES

[1] Placeway, P., Schwartz R., Fung P., and Nguyen, L., 1993, The Estimation of Powerful Language Models from Small and Large corpora, ICASSP-93, Vol. 2, pp. 33-36.

[2] Brown P. F., Della Pietra S. A., Della Pietra V. J., Lai J. C., and Mercer R. L., 1992, An Estimate of an Upper Bound for the Entropy of English, Computational Linguistics, Vol. 18, pp. 31-40.

[3] Chen Standy F. and Goodman Joshua, 1999, An Empirical study of smoothing Techniques for Language Modeling, Computer Speech and Language, Vol. 13, pp. 359-394.

[4] Church K. W. and Gale W. A., 1991, A Comparison of the Enhanced Good-Turing and Deleted Estimation Methods for Estimating Probabilies of English Bigrams, Computer Speech and Language, Vol. 5, pp 19-54.

[5] Dagan I., Maucus S and Markovitch, 1995, Contextual Word Similarity and Estimation from Sparse Data, Computer Speech and Language, Vol. 9, pp. 123-152.

[6] Essen U. and Steinbiss, 1992, Cooccurrence Smoothing for Stochastic Language Modelling, IEEE International conference on Acoustic, Speech and Signal Processing, Vol. 1, pp. 161-164.

[7] Good I. J., 1953, The Population Frequencies of Species and the Estimation of Population Parameters, Biometrika, Vol. 40, pp. 237-264.

[8] Jelinek F., 1997, Automatic Speech Recognition-Statistical Methods, M.I.T.

[9] Jelinek F. and Mercer R. L., 1980, Interpolated Estimation of Markov Source Parameters from Spars Data, Proceedings of the Workshop on Pattern Recognition in Practice, North-Holland, Amsterdam, The Northlands, pp. 381-397.

[10] Juraskey D. and Martin James H., 2000, Speech and Language Processing, Prentice Hall.

[11] Katz S. M., March 1987, Estimation of Probabilities from Sparse Data for the Language Models Component of a Speech Recognizer, IEEE Trans. On Acoustic, Speech and Signal Processing, Vol. ASSP-35, pp. 400-401.

[12] Knerser R. and Ney H., 1995, Improved Backing-Off for M-gram Language Modeling, IEEE International conference on Acoustic, Speech and Signal Processing, pp. 181-184.

[13] Nẩdas A., 1984, Estimation of Probabilities in the Language Model of the IBM Speech Recognition System, IEEE Transactions on Acoustics, Speech, Signals Processing, Vol. 32, No. 4, pp. 859-861.

[14] Nẩdas A., 1985, On Turing's Formula for Word Probabilities, IEEE Trans. On Acoustic, Speech and Signal Processing, Vol. ASSP-33, pp. 1414-1416.

[15] Ney H. and Essen U., 1991, On Smoothing Techniques for Bigram-Based Natural Language Modeling, IEEE International conference on Acoustic, Speech and Signal Processing, pp. 825-828.

[16] Su K. Y., Chiang T. H., Chang J. S., A Overview of Corpus-Based Statistical-Oriented (CBSO) Techniques for Natural Language Processing, Computational Linguistics and Chinese Language Processing, vol. 1, no. 1, pp.101-157, August 1996.

[17] Witten L. H. and Bell T. C., 1991, The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression, IEEE Transaction on Information theory, Vol. 37, No. 4, pp. 1085-1094.

[18] Huang C.-R., 1995, Introduction to the Academic Sinica Balance Corpus, Proceeding of ROCLLING VII, pp. 81-99.

[19] Algort P. H. and Cover T. M., 1988, A Sandwich Proof of the Shannon- McMillan-Breiman Theorem, Ahe Annals of Probability, Vol. 16, No. 2, pp. 899-909.

[20] Jurafsky D. and Martin J. H., 2000, Speech and Language Processing, Prentice Hall, Chapter 6.

[21] Juang B. H and Lo S. H., 1994, On the Bias if the Turing-Good Estimate of Probabilities, IEEE Trans. On Signal Processing, Vol. 42, No. 2, pp. 496-498.

[22] Hermann Ney, Ute Essen, Reinhard Kneser, December 1995, On the Estimation of 'Small' Probabilities by Leaving-One-Out, Vol. 17, No. 12, IEEE PAMI, pp. 1202-1212.

[23] Xuehua Shen, ChengXiang Zhai, Active Feedback in Ad Hoc Information Retrieval, Proceedings of ACM SIGIR 2005.

[24] Seung-Hoon Na, In-Su Kang, Ji-Eun Roh, Jong-Hyeok Lee, An Empirical Study of Query Expansion and Cluster-Based Retrieval in Language Modeling Approach, AIRS 2005.

[25] Guihong Cao, Jian-Yun Nie, Jing Bai, Integrating Word Relationships into Language Models Proceedings of ACM SIGIR 2005

A System for Association Rule Discovery in Emergency Response Data

George Dimitoglou, Shmuel Rotenstreich

Abstract— **Association rule mining is used to find association relationships in data. Our work describes the use of association rule discovery as a basis for creating an early warning bio-terror attack system. The system establishes a baseline of "normal" behavior by mining historical emergency response (911) data. Using probabilistic models, we generate spatial and temporal statistics to correlate incident frequency and location in order to identify if a variation in future incidents carries an outbreak signature consistent with the effects of a biological warfare attack. Using three years of real emergency response data for experimentation, this work is focused on the activities relating to the processing and generation of detection rules. Preliminary results indicate that the system can provide reasonable detection rules but there is also more work to address inherent issues of both emergency response and biological warfare such as data quality during incident reporting and population mobility as it relates to outbreaks.**

Index Terms— Data mining, knowledge discovery, association rules

I. Introduction

EMERGENCY response data is widely used for analysis in various disciplines, ranging from criminal justice and sociology to political science and public health administration. The most common studies in these disciplines evaluate incident frequency and geographical location in search of patterns to reveal the formulation or evolution of trends that could then be addressed with changes in emergency response resources or public policy. This type of "post-mortem analysis" is very beneficial in understanding and profiling typical incident activity such as increased traffic accidents around national holidays or assaults, robberies and murders in crime-ridden certain neighborhoods. An interesting alternative is the use of emergency response data as a tool for incident identification for improved preparedness and prevention. Obviously, this type of use is limited to only a certain type of incidents, mainly those with a fairly prolonged and staged manifestation. Such incidents would be identified within certain symptoms in the population and could be the result of the intentional release of a biological agent or, an industrial accident that caused a toxic spill.

G. Dimitoglou is at the Department of Computer Science, Hood College, Frederick, MD 21701-8575 USA (phone: 301-696-3980; e-mail: dimitoglou@hood.edu).

S. Rotenstreich is at the School of Engineering and Applied Science of the George Washington University, Washington, DC 20052.

This work is focused on using association rule mining to create a bio-terror attack diagnosis system that establishes a baseline of "normal" behavior by mining historical emergency response (911) data. This baseline serves as the threshold of normalcy and any significant deviation from the baseline during "real-time" operations becomes a possible indicator of an attack.

II. Background and System Architecture

Data mining is a widely used analysis tool in many scientific and industrial applications. Association rule mining [1], [6] tries to find association relationships in data. The association relationships are described by rules. Each rule has two measurements: support and confidence. Confidence is a measure of the rule's strength, while support corresponds to statistical significance. Classification is a data mining operation that has been studied extensively in the fields of statistics, pattern recognition, decision theory, machine learning literature, neural network, etc. Clustering [4] is often an important initial step of several in the data mining process. Some of the data mining approaches which use clustering are *database segmentation*, *predictive modeling*, and *visualization* of large data-bases [7]. Typical pattern clustering activity involves the following [7]: (a) Pattern representation (optionally including feature extraction and/or selection), (b) Definition of a pattern proximity measure appropriate to the data domain, (c) Clustering or grouping, (d) Data abstraction (as needed), and (e) Assessment of output (as needed). Sequential pattern mining is the process to analyze a collection of data over a period of time to identify trends [1]. Sequential pattern mining is closely related to association rule mining, except that the events are linked by time.

To mine, process and use 911 response data, we developed a software bio-terror attack diagnosis system. The main function for this system is to generate association rules from an incident response database and second, to attempt to identify a geographic pattern in the instances. The rationale behind the system is by identifying the occurrence of certain symptoms during a certain time period in a geographic area may be an indication of a health anomaly pattern within the population (i.e. a virus spreading).

The system uses 911 call data as input and generates (a) a set of baseline rules that reflect normal conditions and (b) alerts

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 193–199.

that are triggered when new, incoming calls deviate from the existing baseline rules. The intention is for these rules to be unambiguously formulated and easily readable so it is easy to be either parsed electronically or read by humans. For example, the following, although simplistic rule:

IF respiratory incidents > 20 per month in area *G* THEN alert;

accommodates both requirements first, by being unambiguous as to differentiating the condition-action parts of the rule and second, by being easily readable so anyone can understand there is action to be taken.

Figure 1 provides an overview of the system architecture. The system contains three processing engines and four repositories.

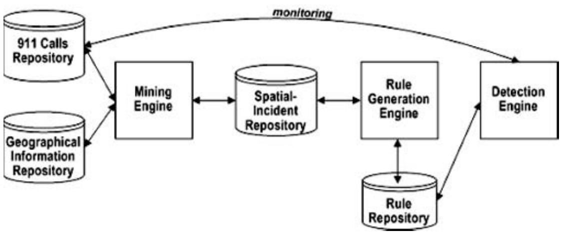


Figure 1: Bio-terror detection system architecture.

The system is comprised of:

- A Mining Engine processes the 911 calls stored in the Calls Repository and generates association rules that are in turn correlated with the contents from a Geographical Information Repository [7]. The Mining Engine generates data and populates a Spatial-Incident Repository by statistically, methodically and mathematically trying to identify unifying patterns and conditions
- A Rule Generation Engine processes the generated association rules and generates baseline rules that are stored in a Rule Repository. Baseline rules are the rules that have been identified by the data mining computations to reflect as the potential occurrence of bioterrorism events.
- The Detection Engine works as a monitor that constantly examines in real-time incoming 911 calls and applies the relevant rules from the Rule Repository.

III. METHODOLOGY AND EXPERIMENTATION

Using data from Regional Emergency Medical Communication Systems (REMCS), the 911 Calls Repository is populated with entries reflecting dispatch requests. The 911 REMCS Call Center receives an average of approximately 15,000 calls per year. The dataset used in our study contained 30,664 calls for a two-year period between January 1997 to December 1998. A sample segment of the data set is displayed in Table 1.

The columns in Table 1 contain information such as an internal identification number (**CALL.ID**), the date (**DATE.REC**), address (**LOCATION**), grid location information (**GRID**) and the patient's condition description (**PAT.COND.DESCR**).

CALL.ID	DATE.REC	LOCATION	GRID	PAT.COND.DESCR
57644	01/01/02	129 WRIGHT ST-NK.3 FL	516	BREATHING PROBS/ABNORM BREATHI
57645	01/01/02	748 S 10 ST-NK	512	TRAUMA/POS DANGEROUS BODY AREA
57646	01/01/02	354 PARK AV-NK.305	212B	PREGNANCY/2nd TRIMESTER BLEED
57647	1/1/2002	100 RT 1&9-NK SOUTH	189	TRAFFIC ACCIDENT W/ INJURIES
57648	01/01/02	MERCHANT ST-	316	HEMORRHAGE/POS DANGEROUS BLEED
:	:	:	:	:
:	:	:	:	:
86578	01/01/02	182 COURT ST-NK	411	SICK PERSON/NO PRIORITY SYMPTO

Table 1:Sample Data

The REMCS system has a grid designation that provides information about the general area of the incident location. Analysis allows us to view incidents grouped within the confines of a larger geographical area and patterns of incidents may also be diagnosed in relation to incidents in adjacent grids. Figure 2 illustrates how mock grid information is overlaid on top of a Newark area map. Patient address information is matched with the corresponding grid number during the response call in order to provide information for ambulance or emergency medical staff deployment.

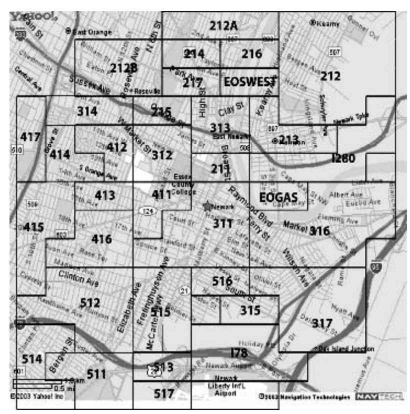


Figure 2: Conceptual Map Overlay. The grid overlay for a metropolitan area served by the REMCS.

A. Process Flow

Figure 3 illustrates the detection rule generation pipeline methodology. Raw data is pre-processed to remove entries that are known to have no relevance when mining for biological warfare incidents (i.e. traffic accidents). The Space-Time Permutation Analysis processing generates probability models that correlate and contain spatial and temporal statistics. The Rule Generation Engine receives the probabilistic models and converts them to event-condition-action (ECA) rules [5]. Finally, these ECA rules are farther processed to identify those rules that provide a higher level of confidence to generate the most quality, definitive detection set of rules.

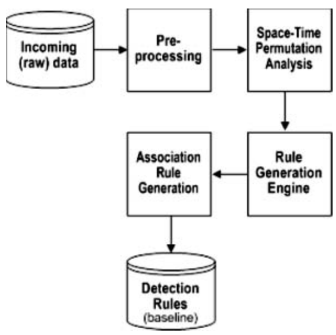


Figure 3: The Detection Rule process flow

B. Pre-processing and Space-Time Permutation Analysis

First, the quality and appropriateness of the dataset is considered. The given dataset provides numerous types of calls ranging from traffic accidents to cardiac arrests. Therefore, due to the multitude of causes for 911 calls, some pre-processing is required to eliminate unrelated causes to bio-terror detection (e.g., "traffic accident" and "assault") and concentrate on symptoms such as "respiratory breathing problem", "fever" and "non-traumatic abdominal pain". In addition, more data must be removed because the raw dataset contains data such as the dispatcher information, ambulance and personnel dispatched, which are not only relevant to our analysis, but they also introduce "noise".

Second, we process the data using SaTScan [9][10]. SaTScan is based on the space-time permutation model using data (instead of the entire dataset), with location and date information for each case. The number of observed cases in a cluster is compared to what would have been expected if the spatial and temporal locations of all cases were independent of each other so that there is no space-time interaction. Therefore, there is a cluster in a geographical area if, during a specific time period, that area has a high proportion of excess cases or a smaller deficiency of cases than surrounding areas. So if one geographical area has twice the number of cases while other areas have a normal amount of cases, then there will be a cluster in that first area. This model automatically adjusts for both purely spatial and purely temporal clusters. Space-time permutation clusters may be due either to an increased risk of disease, or to different geographical population distribution at different times, where for example the population in some areas grows faster than in others.

Using SaTScan we perform a prospective Space-Time analysis scanning for clusters with high rates using the Space-Time Permutation model for a study period between 1997/1/1-1998/12/30. The output of the study yields the results in the table below for a *most likely* and *secondary clusters*.

	MOST LIKELY CLUSTER	SECONDARY CLUSTERS
Location IDs included	68, 316, 66, 67, 212, 311, 213, 267, 211, 315, 313, 515, 514, 516, 317, 415, 511, 414, 517, 513, 416, 417, 413 68, 316, 66, 67, 212, 311, 213, 267, 211, 315, 313, 314, 412	
Coordinates / radius	(40.725491 N, 74.204376 W)/1.64 km	(40.738716 N, 74.114983 W)/3.17 km
Time frame	1997/1/1-1998/12/301997/1/1-1998/12/30	
Number of cases	561 (506.31 expected)	363 (330.62 expected)
Overall relative risk	1.1351.085	
Test statistic	2.36036 1,74378	
Monte Carlo rank	100/1000	981/1000
P-value	0.100	0.981
Null Occurrence	Once in 12 days	Once in 7 days
The test statistic value required for an observed cluster to be significant at level: 0.01: 3.463243 and 0.05: 2.665597		

Table 2: SaTScan output for most likely and secondary clusters

The output from the space-time permutation analysis (Table 2) is sent to the Rule Generation Engine where the output is parsed and parameters are being extracted and processed to generate IF-THEN type detection rules.

IF Symptom=breathing problems AND location_id={417}AND number_of_cases > 561 THEN alert
IF Symptom=breathing problems AND location_id={316}AND number_of_cases > 561THEN alert.
IF Symptom=breathing problems AND location_id={517}AND number_of_cases > 561THEN alert.
IF Symptom=breathing problems AND location_id={513} AND number_of_cases > 561 THEN alert.

Generation of alerts is simplified by detection rules. They culminate in counting cases of breathing problems in different regions and checking against the threshold. Since 911 services are continuous (24x7x365) this part of the process can be run frequently to adjust to changing conditions, thereby indirectly adjusting the detection rules.

C. Association Rule Discovery

First, there is no change or special requirements for the association rule discovery at the pre-processing stage beyond the pre-processing performed before (section B). Consequently, for input, the same dataset as before is used to generate the association rules.

Association rules show how attribute value conditions or variables are correlated (vary or behave alike) in a given dataset. A common use of association rule mining is Market Basket Analysis [3]. Typical examples of Market Based Analysis come from retail sales where marketing and inventory-control efforts can benefit by identifying if certain groups of products are consistently purchased together. This analysis is performed by selecting a certain collection of variables that hint simultaneous variability may indicate a pattern. Association rules can provide information in the form of "if-then" rule statements. These rules are generated from the data and -unlike the "IF-THEN" logic rules- association rules are probabilistic in nature. So, in addition to the predicate and result parts, an association rule has two numbers that express the degree of uncertainty about the rule. In association analysis the predicate and result components of a rule are sets of items (called *itemsets*) that are disjoint (do not have any items in common). The first number is the *support* for the rule. The second number is the *confidence* of the rule.

Definition 1: The *support* is the number of transactions that include all items in the predicate and result parts of the rule and *support* is sometimes expressed as a percentage of the total number of records in the repository.

Definition 2: *Confidence* is the ratio of the number of transactions that include all items in the result part of the rule as well as the predicate (namely, the *support*) to the number of transactions that include all items in the predicate.

Unlike typical logic-based rules, where the rules are given and they must be applied to tasks or activities, we performed Link Analysis [1] to discover the association rules from the data set, in the form: IF <*LHS*> THEN <*RHS*>
where LHS stands for the left-hand-side of the rule (the predicate) and RHS stands for the right-hand-side of the rule, the result. To ensure that discovered rules are meaningful and strongly related instead of just not merely related. We use the Optimum Search for Unsorted Search (OPUS) software and algorithm [19], [20] which finds the rules that maximize an arbitrary function measuring rule quality. The rule quality is based on:

(a) *Prevalence* (*lift* or *leverage*). The probability that LHS and RHS occur together.
(b) *Predictability* (*confidence*). The probability of RHS given LHS.

The search algorithm is the following:

```
OPUS_AR(CurrentLHS, AvailableLHS, AvailableRHS)
So Far := {}
foreach P in AvailableLHS
   NewLHS := AvailableLHS – P
   AvailableLHS :=AvailableLHS – P
   IF cover(NewLHS) < minLHScover. THEN
    NewAvailableRHS = AvailableRHS
    FOR EACH Q in AvilableRHS
       IF Strength > minStrength and able to fit into m best rules THEN
         Record NewLHS Q
         IF any RHS condition Q for which
            cover(NewLHS U {Q}) < minRHScover OR
            opt_strength < min_strength OR opt_lift < min_lift THEN
            NewAvailableRHS := NewAvailableRHS – Q
            If NewAvailableRHS != {} THEN
               OPUS_AR(NewLHS, SoFar, NewAvailableRHS)
               SoFar := SoFar U {P}
```

Algorithm 1: The OPUS algorithm pseudocode for rule association discovery

D. Results

The results identify 645 cases with 72 items, generating 36 values allowed on the LHS and 36 values allowed on RHS. Multiple rules are being generated with different symptoms but again we concentrated on breathing problem symptoms.

The generated association rules can be classified under two general categories. The first category contains association rules that relate the symptom "Breathing Problem" with particular grid locations in the covered metropolitan area (Table 3). The second category contains association rules that relate the symptom "Breathing Problem" with a particular grid location and a temporal period (Table 4).

For example, the first association rule from Table 4 (IF *breathing problem* THEN *grid=316)* reveals that there is an association between the symptom "*breathing problem*" attribute and its appearance in another attribute (grid area 416). The frequency and relation between those attributes is expressed with numerical values. *Coverage* is the proportion of data covered by the LHS, *Support* is the proportion of data jointly covered by the LHS and RHS and *Strength* is the proportion of data covered both by the LHS and the RHS. *Lift* is a composite calculation of the *Strength*, divided by the proportion of all data that are covered by the RHS.

Finally, *Leverage* describes the proportion of additional data covered by both the LHS and RHS above those expected if the LHS and RHS were independent of each other. In our case, the values of the additional parameters (*Coverage*, *Support*, *Strength*, *Lift*, *Leverage*) are interpreted as follows:

	Association Rules	Coverage	Support	Strength	Lift	Leverage
1	IF symptom=Breathing Problem THEN grid_area=316	0.158	0.009	0.057	1.84	0.0041
2	IF symptom=Breathing Problem THEN grid_area=516	0.158	0.01	0.063	1.81	0.0045
3	IF symptom=Breathing Problem THEN grid_area=312	0.154	0.018	0.117	1.62	0.0069
4	IF symptom=Breathing Problem THEN grid_area=417	0.154	0.01	0.065	1.8	0.0045
5	IF symptom=Breathing Problem THEN grid_area=311	0.151	0.015	0.099	1.95	0.0073
6	IF grid_area=311 THEN symptom=Breathing Problem	0.051	0.015	0.294	1.95	0.0073
7	IF grid_area=314 THEN symptom=Breathing Problem	0.033	0.009	0.273	1.77	0.0039
:	:	:	:	:	:	:
8	IF grid_area=214 THEN symptom=Breathing Problem	0.033	0.009	0.273	1.77	0.0039

Table 3: Relevant association rules to the "breathing problem" symptoms and grid areas

	Association Rules	Coverage	Support	Strength	Lift	Leverage
1	IF symptom=Breathing Problem AND date=4th week of Sept THEN grid_area=511	0.004	0.003	0.75	20.27	0.0029
2	IF symptom=Breathing Problem AND date=3rd week of Dec THEN grid_area=516	0.012	0.008	0.667	22.95	0.0079
3	IF symptom=Breathing Problem AND date=3rd week of Dec THEN grid_area=311	0.017	0.008	0.5	7.09	0.0071
4	IF symptom=Breathing Problem AND date=4th week of Sept THEN grid_area=511	0.004	0.003	0.75	20.27	0.0029
5	IF symptom=Breathing Problem AND date=3rd week of Dec THEN grid_area=516	0.012	0.008	0.667	22.95	0.0079
6	IF symptom=Breathing Problem AND date=4th week of Dec THEN grid_area=311	0.017	0.008	0.5	7.09	0.0071
7	IF symptom=Breathing Problem AND grid_area=511 THEN date=3rd week of Dec	0.006	0.003	0.5	23.81	0.0029
:	**:**	**:**	**:**	**:**	**:**	**:**
8	IF symptom=Breathing Problem AND date=4th week of Sept THEN grid_area=511	0.004	0.003	0.75	20.27	0.0029

Table 4: Association rules with attributes: symptoms, geographical (grid location) and temporal information (date). Rule attribute parameters (Coverage, Support, Leverage, Strength Lift) are omitted.

(a) *Coverage*. There are 0.158 cases indicate "*breathing problem*" as symptoms.
(b) *Support*. 0.009 of cases indicate "*breathing problem*" symptoms together with grid area 416.
(c) *Strength*. 0.057 of the cases with "*breathing problem*" symptoms also appear in grid area 416.
(d) *Lift*. For cases with this symptom, the frequency of this symptom is 1.84 times normal.
(e) *Leverage*. As a result, 0.0041 of all cases have this kind of symptom than would be expected if there were no association between these attributes.

A. Generation of Detection Rules

The generation of the Detection Rules is a synthesis of the results from the space-time permutation analysis and the association rule discovery techniques. The output of the space-time permutation statistics are correlated with the association rules to produce detection rules with the highest confidence. The detection rules in this case reflect the baseline of the system as the given data set was supplied during a period were there were no indications for any contamination that would be the root cause of the reported "breathing problems". Following, there is a sample listing (5%) of the total number (over 200) of detection rules generated from the sample execution (Figure 4).

1. IF symptom=Breathing Problem AND date=4th week of Dec AND grid_area=311 AND year_to_date_cases > 561 THEN alert
2. IF symptom=Breathing Problem AND grid_area=417 AND year_to_date_cases > 561 THEN alert
3. IF symptom=Breathing Problem AND date=3rd week of Dec AND grid_area=516 AND year_to_date_cases > 561 THEN alert
4. IF symptom=Breathing Problem AND grid_area=516 AND year_to_date_cases

> 561 THEN alert
5. IF symptom=Breathing Problem AND date=4th week of Sept AND grid_area=511 AND year_to_date_cases > 561 THEN alert
6. IF symptom=Breathing Problem AND date=4th week of Sept AND grid_area=511 AND year_to_date_cases > 561 THEN alert
7. IF grid_area=311 AND symptom=Breathing Problem AND year_to_date_cases > 561 THEN alert
8. IF grid_area=314 AND symptom=Breathing Problem AND year_to_date_cases > 561 THEN alert
9. IF symptom=Breathing Problem AND date=4th week of Sept AND grid_area=511 AND year_to_date_cases > 561 THEN alert
10. IF symptom=Breathing Problem AND grid_area=511 AND date=3rd week of Dec AND year_to_date_cases > 561 THEN alert

Figure 4: Generated Detection Rules.

This sample of detection rules along with the entire collection of rules generated from this study belong to the most likely cluster of results and also have the highest confidence from the association rules. However, this does not mean that more detection rules can not be generated. In fact, the number of generated detection rules from this study is over 200 and both the parameters of the rules and the detection rules themselves can be reduced or increased depending on how responsive (conservative or aggressive) the system needs to be in identifying patterns of anomalous behavior. This level of responsiveness may be given to the authorities in charge of bio-terrorism defense in ways to allow them suitable controls. For instance, under escalated alert levels, more control may be sought and the controllers may adjust detection to that level.

B. *Output Visualization*

Next, we highlight those grid areas that the rules have indicated a strong correlation between the symptoms and the areas. From there, either visually or automatically patterns may be identified. Automatically, the grid may be translated to an adjacency matrix and then parsed for those areas with common sides (rule applicability).

This processing can be performed either statically (daily, weekly or monthly) or dynamically (constantly running), generating rules as patient calls arrive. While the latter may be computationally more expensive, the OPUS algorithm is exponential depending on the data storage [19].

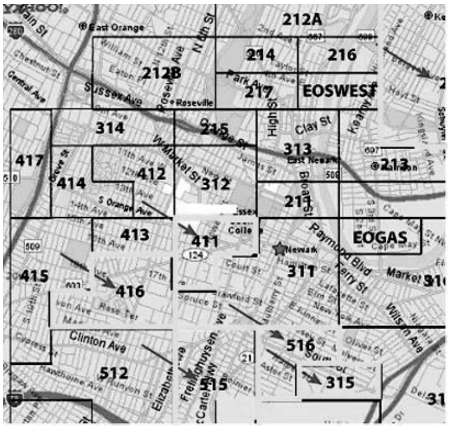


Figure 5: Grid overlay with highlighted areas were symptoms generate strongly supported association rules

From the above rules, if we highlight the grid information on the map overlay (Figure 2), is obvious that there is some identifiable pattern of incidents South and West of the city center. This is a preliminary result as at this point further analysis is required to determine if there is indeed a pattern of a contamination that affects the population (hence increased emergency calls of a certain type) or, this result is just a coincidence.

IV. Conclusion and Future Work

We have demonstrated that relationships can be discovered automatically from a set of emergency response data and expressed as association rules. Using data from a Regional Emergency Medical Communication System (REMCS) we are able to generate association rules that may serve to identify patterns of health anomalies in the population in case of a biological warfare attack. The detection rules are very flexible constructs that can provide great flexibility in expressing even more complex detection parameters and relation between attributes. An example of a powerful rule detection type are the Computation Detection Rules that include computations (i.e. calculations of environmental sensor measurements embedded in the data) to derive threshold parameters for detection. One shortcoming inherent to emergency response data is data quality. The condition description for an incident is usually a general, standard description selected by the dispatcher who culminates the described symptoms of either a distressed patient or an equally anxious bystander who places the call. Therefore, it is possible that there may be numerous instances where the patient condition description is either misunderstood or exaggerated. Another issue is that there are also additional dimensions of this dataset not covered by the detection system at this point. The temporal relation of the rules for example, is not strongly related with the generated rules. To that end, it would be interesting to determine a possible correlation between the timing (the

This page intentionally blank

epidemic curve) of the incidents and the particular types of incidents in order to remove any coincidental conclusions and further refine the results. If for example the detection engine provides alerts for incidents that carry an anthrax attack for an area but over a five-week period, it is obvious that this is an erroneous result as the incubation period of anthrax spores is 1-3 days. This type of contamination-specific characteristics (e.g. expected lifetime of a virus) may further refine the results. Finally, another limitation of this analysis is the lack of consideration for population mobility. As shown in population mobility studies in other domains [13], it is also very likely that people may move through multiple grid sections (i.e. commuting to work, going to school) multiple times even within a single day. Therefore it is difficult to determine if some of the incidents are not localized to the particular geographical area but they are cause by transient behavior from the population.

REFERENCES

[1] Agrawal R., Imielinski, T., Swami A.N. (1993), Mining Association Rules between Sets of Items in Large Databases, Proceedings of the 1993 {ACM} SIGMOD International Conference on Management of Data, 26—28, Washington, D.C., Eds. Peter Buneman and Sushil Jajodia, pp.207—216.

[2] Agrawal, R., Mannila, S., H., Toivonen, H. and Verkamo, A.I. (1996) Fast discovery of association rules. In U. Fayyad et al., editors, Advances in Knowledge Discovery and Data Mining, 307-328. AAAI Press, 1996.

[3] Brin, S., Motwani, R., Silverstein, C. (1997) Beyond Market Baskets: Generalizing Association Rules to Correlations. 265-276 *Electronic Edition* (ACM DL).

[4] Chen, M-S, Han, J., Yu, P.S. (1996) Data mining: an overview from a database perspective, IEEE Trans. On Knowledge and Data Engineering, Vol. 8, pp. 866—883.

[5] Gatziu, S. Dittrich, K.R.,(1993) Events in an Active Object-Oriented Database System, Proc. 1st Intl. Workshop on Rules in Database Systems (RIDS), Springer-Verlag, Workshops in Computing, Edinburgh, UK, Eds. Paton, N.W. and Williams, H.W.

[6] Han J., Fu, Y. (1995), Discovery of Multiple-Level Association Rules from Large Databases, Proc. of 1995 Int'l Conf. on Very Large Data Bases (VLDB'95), Zurich, Switzerland, September 1995, pp. 420—431.

[7] Jain, A.K., M.N. MURTY, P.J. FLYNN, Data Clustering: A Review, ACM Computing Surveys, Vol. 31, No. 3, September 1999.

[8] Korte, G. (2000) The GIS Book, 5th edition, OnWord Press.

[9] Kulldorff, M. (2003). A Tree-Based Scan Statistic for Database Disease Surveillance, 2003.

[10] Kulldorff M, Heffernan R, Hartman J, Assunção RM, Mostashari F. A space-time permutation scan statistic for the early detection of disease outbreaks. PLoS Medicine, 2:216-224, 2005.

[11] Kusiak Andrew, Shital Shah, and Bradley Dixon , Data Mining Based Decision-Making Approach for Predicting Survival of Kidney Dialysis Patients, In D.D. Feng And E.R. Carson (Eds), Modeling And Control on Biomedical Systems 2003, Proceedings of the IFAC 2003 Symposium on Modeling and Control of Biomedical Systems, Melbourne, Australia, Elsevier, Amsterdam, The Netherlands, August 2003, pp. 35-39.Mandl, K. D. (2001). Infrastructure and Methods to Support Real Time Bio-surveillance, 2001.

[12] Magnum Opus - Software, Copyright (c) 1999, 2000, 2001 G. I. Webb & Associates Pty Ltd.

[13] Mostashari F, Kulldorff M, Hartman JJ, Miller JR, Kulasekera V. Dead bird clustering: A potential early warning system for West Nile virus activity. Emerging Infectious Diseases, 9:641-646, 2003.

[14] Nobutaka, N. Hishigaki, H. Takagi, T. (1996) Application of Data Mining to Genetic Linkage Analysis, Genome Informatics 7: 266-267.

[15] Nordin J, Goodman M, Kulldorff M, Ritzwoller D, Abrams A, Kleinman K, Levitt MJ, Donahue J, Platt R. Using modeled anthrax attacks on the Mall of America to assess sensitivity of syndromic surveillance. Emerging Infectious Diseases, 11:1394-1398, 2005.

[16] Smyth , Padhraic, (2003) Probabilistic Knowledge Discovery and Data Mining: An Integrated Approach at the Interface of Computer Science and Statistics Information and Computer Science University of California, Irvine.

[17] Song C, Kulldorff M. Power evaluation of disease clustering tests. International Journal of Health Geographics, 2:9, 2003.

[18] Srikant, R. Agrawal, R. (1997) Mining generalized association rules, Future Generation Computer Systems, Vol. 13, 2-3, pp. 161--180

[19] Webb, G.I., (1995) OPUS:An Efficient Admissible Algorithm for Unordered Search, Journal of Artificial Intelligence Research,3, pp. 431-465, 1995.

[20] Webb, G.I., (2000) Efficient search for association rules. In Proc. KDD-2000, Knowledge Discovery and Data Mining, pp. 99-107, August 2000, Boston.

George Dimitoglou is an Assistant Professor of Computer Science at Hood College, Frederick, MD. He is a member of the ACM and the IEEE.

Shmuel Rotenstreich is an Associate Professor of Computer Science at the School of Engineering and Applied Science of The George Washington University, Washington, DC.

Towards Logarithmic Search Time Complexity for R-Trees

Richard Göbel
University of Applied Sciences Hof, Alfons-Goppel-Platz 1,
95028 Hof, Germany

***Abstract*-Index structures are frequently used to reduce search times in large databases. With index structures like the B-Tree search time grows only logarithmic with the size of a database for several types of searches. This means that the search time is almost constant for very large database systems even if their size grows significantly. Conventional index structures however do not well support searches specifying lower and/or upper bounds for more than one attribute (multidimensional range searches). Therefore R-Trees are increasingly used in this application context. A typical application domain of R-Trees are spatial database systems with two dimensional search conditions specifying upper and lower bounds for longitude and latitude values. Unfortunately R-Tree efficiency does not meet the expectations in many cases. Theoretical analysis of this problem showed, that search time grows much faster than logarithmic for two and more dimensional range searches in contrary to the one dimensional case. In this paper we prove that a logarithmic search complexity can be achieved for two dimensions, if the form of nodes is optimized relative to the form of search conditions. Based on this result, the paper investigates the form of nodes generated by different existing tree packing methods. Since existing methods fail to ensure the required form, a new tree packing method is proposed which improves the chance to meet the identified requirements.**

I INTRODUCTION

Index structures help to reduce search time for database systems. Without an index structure search time grows at least linearly with the number of "relevant" entries in the database. With an index structure it is possible to navigate to the search result instead of checking every individual entry. A frequently used index structure is the B-Tree [9]. It ensures a time complexity growing logarithmic with the number n of relevant entries for some queries. As an example, queries specifying a lower and an upper bound for a single column can be efficiently supported by a B-Tree (e.g. find all persons with an income between 50 k and 100 k). So called multi-dimensional range searches, specifying upper and/or lower bounds for more than one column however cannot be directly supported by the B-Tree (e.g. persons with an age between 18 and 21 and with an income between 50 k and 100 k). Different index structures have been proposed since the seventies (e.g. [4], [5], [6], [15]) for this type of search conditions. An overview on "Multidimensional Access Methods" is given by [12].

Multidimensional access methods can be divided into replicating (e.g. [6]) and non-replicating index structures (e.g. [15]). Replicating index structures ensure a logarithmic time complexity but their space requirements make it difficult to use them in real applications. On the other hand non-replicating index structures have moderate space requirements facilitating their application even in very large databases. Unfortunately theoretical analysis identified $\Omega(n^{(d-1)/d})$ (d number of columns/dimensions) as a lower bound for the time complexity of this type of index structures ([20], [18]).

During the nineties interest in multidimensional range searches increased significantly due to database applications in the domains Multimedia, Bioinformatics and Geographical Information Systems (GIS). Of special interest were spatial database systems as the backend for GIS, because almost all major database providers offer a spatial extension for their database engine. Many of them use the R-Tree ([15]) as a multidimensional index structure for supporting spatial searches. As a consequence research has investigated the R-Tree in more details and proposed several extensions and improvements (e.g. [24], [8], [7], [2], [10]). Of particular interest are so called tree packaging methods which generate a tree from a given set of entries (e.g. [23], [17], [19] and [13]). These tree packing methods improve the chance to optimize a tree according to some selected criteria. Although all proposed methods cannot be better than the identified lower bound, these methods might provide better results in the average case than incrementally generated R-Trees. On the other hand most of the mentioned methods may have significant longer search times in several cases than the identified lower bound. Only most recently a new tree packing method was proposed which guarantees this lower bound also as the upper bound for all searches [1].

In comparison to a logarithmic performance the lower bound $\Omega(n^{(d-1)/d})$ is probably not acceptable for very large databases (see for example [14], [11]). Logarithmic performance means that response times are almost constant for very large databases, if some entries are added.

Often experiments show better search times than the identified lower bound. Some theoretical analysis is available as well ([25], [16]). Unfortunately this analysis does not clearly identify conditions which can be easily checked for an application.

In general the efficiency of a database application based on R-Trees is not clear. Currently an application programmer takes the risk that database efficiency decreases significantly if the amount of data grows or changes. This paper addresses this gap by providing conditions under which efficient R-Tree support is guaranteed. These conditions refer to the form of nodes and search conditions and can be easily checked for a database.

The paper starts with a theoretical analysis of R-Tree performance in section II. Section III introduces a new tree packing method considering the identified conditions. The new method is applied to test data in section IV and compared with other tree packing methods. Section V summarizes the results.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 201–206.

II Analysis

This section investigates the dependency of the number of visited nodes from the number of entries n and the size of the result set m (number of entries satisfying the search condition). For this purpose the section investigates the number of overlaps between a search condition and the nodes of an arbitrary level (depth) in a tree (e.g. all leaf nodes). This number multiplied with the depth of the tree provides an upper bound for the total number of nodes which need to be visited.

In this section we focus on two dimensional R-Trees. All nodes of a two dimensional R-Tree can be considered as rectangles in two dimensional space, where the sorted value spaces of the involved columns represent the two dimensions. A rectangle defines lower and upper bounds for both dimensions. A rectangle of a parent node includes all rectangles of its child nodes and the leaf nodes contain entries as points. Note that for an R-Tree all rectangles are minimal (minimal bounding rectangle). A search condition is also a rectangle (search rectangle). A search starts at the root node and continues recursively at all child nodes which overlap the search rectangle. On the leaf level all points are checked for the search condition.

It is probably obvious that the number of visited nodes grows at least linearly with the size m of the result set. The dependency between the number of entries and the visited nodes is however not obvious. In fact a search rectangle may overlap a large number of nodes without retrieving any results. If however every node contributes at least one entry to the search result, then the size of the result set m is an upper bound for the number of visited nodes (not more than m nodes need to be searched). This also means that the number of visited nodes would not be dependent on the total number of entries n.

Accordingly we will start our analysis by identifying cases of overlaps which contribute at least one entry to a search condition. Since we cannot avoid other types of overlaps we will derive conditions which limit overlaps not contributing entries.

For this analysis we distinguish between strong and weak overlaps. Two nodes weakly overlap, if borders of these two nodes just meet (common points are on borders of nodes). A strong overlap is given, if two nodes share common inner points. Figure 1 shows examples of weak and strong overlaps.

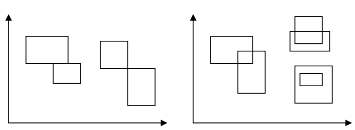


Figure 1: Weak (left) and strong overlaps (right)

Our analysis is based on the following general assumptions:

- For the rest of the paper we assume that nodes on a given level do not strongly overlap. This property can be guaranteed by tree packing methods for point data.
- For technical reasons we assume that lower bounds of ranges are always less (not equal) than upper bounds (arbitrarily small differences are still possible).

In the next section we will discuss a tree packing method satisfying these conditions facilitating efficient searching.

We start the analysis by identifying a condition under which a node contributes at least a single entry to a search region. It is probably obvious that a node contributes all its points if the node is fully included in the search region. In the two dimensional case it is already sufficient if one edge is included in the search region because an edge of a minimal bounding rectangle includes at least one point (otherwise the bounding rectangle would not be minimal). Figure 2 shows such an overlap between a search region (bounded by a dotted line) and a tree node (bounded by a solid line).

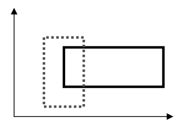


Figure2: Edge of node included in search region

Lemma 1 In a two dimensional R-Tree a node contributes a point to a search region if the search region contains at least an edge of the node.

All other cases of overlaps may not always contribute a point to a search result. There are however other cases for which the set of overlapping nodes can be limited to a constant number not depending on the number of entries in the database. One of these cases deals with the situation that nodes overlap a corner point of a search region.

The first question in this context is how many nodes may overlap a point if we have no strong overlaps between these nodes. We use Figure 3 to analyze the worst case. This diagram splits the neighborhood of a point into four different quadrants (**n**orth-**w**est, **n**orth-**e**ast, **s**outh-**w**est, **s**outh-**e**ast).

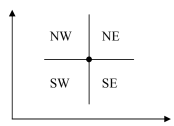


Figure3: Quadrants in neighborhood of point

A single node overlapping a point blocks:

- all quadrants if the point is inside the node (not on edge).
- two quadrants if the point is on an edge.
- a single quadrant if the point is the corner of a node

for further overlaps. As a consequence a point may be overlapped by at maximum 4 different nodes.

Lemma 2 Given is a set of two dimensional *NODES* without strong overlaps. Then a single point is overlapped by not more than 4 nodes.

A search rectangle has four corner points. This means that not more than 16 nodes overlap the corners of a search rectangle.

Lemma 3 Given is a set of two dimensional *NODES* without strong overlaps. Then the search region is overlapped by not more than 16 nodes which include corners of the search region.

Figure 4 summarizes the results we got so far. For this purpose it summarizes all possible overlaps between a node (solid line) with the bounds ($[l_1,u_1],[l_2,u_2]$) and a search region (dotted line) with the bounds ($[v_1,w_1],[v_2,w_2]$) in two dimensions by comparing their lower bounds as well as their upper bounds.

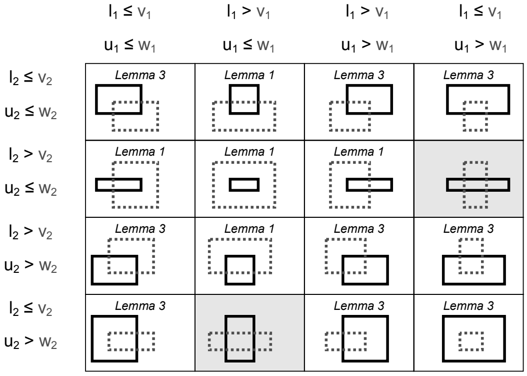


Figure 4: Types of Overlaps

It is interesting to see that just two cases are missing. In both cases search regions and nodes have a very different form (one long and narrow, other short and wide). This means that we have a type of "incompatibility" between them.

This incompatibility could be avoided if we assume that nodes and search regions have exactly the same ratio between width and height. This is however an unrealistic requirement because it would be very difficult or even impossible to generate a tree where all nodes have exactly a predefined ratio. Even worth such an R-Tree would only support searches well which have exactly the given ratio.

To avoid this problem we accept a certain degree of deviation for the ratio between width and height. This deviation is given by the factor k. We may choose this factor freely but it needs to be fixed for the generation of a tree. A large k allows a larger deviation and makes it easier to generate an R-Tree meeting our requirements. On the other hand a larger k will also result in a larger upper bound for the number of visited nodes reducing the efficiency for a search.

The factor k is used as follows:

- If a node is greater or equal in one dimension than the search region
- then it shall be not more than k times less in the other dimension.

As an example if the width of the node (w_n) is greater than the width of the search region (w_s) then the height of the node shall not be more than k times less than the search region:

$h_n * k > h_s$

This condition means that at maximum $k+2$ nodes are needed to overlap the search rectangle vertically and not more than 3 nodes are needed to cover it horizontally. In total the search rectangle cannot be overlapped by more than $(k+2)*3$ nodes of this type. Figure 5 shows the maximum number of nodes overlapping a search rectangle for $k = 2$.

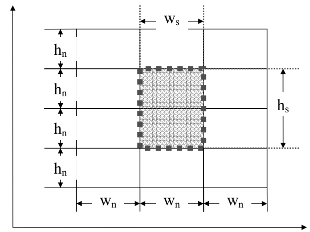


Figure 5: Nodes overlapping a search rectangle

With a similar argument as before we get $(k+2)*3$ as the upper bound for the number of overlaps if the height of the node (h_n) is greater than the height of the search region (h_s). Now all these results are put together in the following theorem. The theorem shows that the number of visited nodes just grows linearly with the size of the result set (points included in search range) and not at all with the number of entries.

Theorem Let *NODES* be a set of *two dimensional* nodes overlapping a search rectangle. The size of the search result (number of points contained in search rectangle) is m. Further if a node is greater than the search region in one dimension than the search region needs to be less than k times greater in the other dimension.

In this case the number of nodes is limited as follows:

$|NODES| \le m+6*k+28$

This theorem follows from the fact that the set *NODES* can be split into 4 subsets such that every node is contained in at least one of these subsets:

- $NODES_1$: corner overlap (Lemma 3)
- $NODES_2$: "side" fully included in search region (Lemma 1)
- $NODES_3$: width of node greater than width of search region.
- $NODES_4$: height of node is greater than height of search region

According to our previous results we have an upper bound for every of these subsets:

$$|\boldsymbol{NODES}| = |NODES_1 \cup NODES_2 \cup NODES_3 \cup NODES_4| \le |NODES_1| \cup |NODES_2| \cup |NODES_3| \cup |NODES_4| \le 16+m+3*(k+2)+3*(k+2) \le \boldsymbol{m+6*k+28}$$

The precondition of Theorem 1 defines requirements for the ratio of a node in the search tree. The following lemma rephrases this condition by providing a valid range for the height of the node depending on its width, the factor k and the ratio between the height and the width of the search rectangle.

Lemma 5 Let $r=h_s/w_s$ be the ratio between the height and width of a search rectangle Furthermore we assume the following condition for every node in the set *NODES*:

$$w_n * r / k < h_n < w_n * k * r$$

Then the preconditions of theorem 1 are satisfied for this node:

$$w_n > w_s \Rightarrow k * h_n > h_s \qquad \text{(i)}$$

$$h_n > h_s \Rightarrow k * w_n > w_s \qquad \text{(ii)}$$

This lemma follows by applying simple transformations to the preconditions. In the first step we use $w_n*r/k < h_n$ (precondition of lemma) and the definition of the ratio $r = h_s / w_s$ to prove (i):

$w_n > w_s$

$\Rightarrow w_n*r/k > w_s*r/k \Rightarrow h_n > w_n*r/k > w_s*r/k \Rightarrow h_n > h_s*k \Rightarrow h_n/k > h_s$

In a similar way we prove (ii) by using $h_n < w_n*k*r$ and the definition of the ratio r:

$h_n > h_s \Rightarrow w_n*k*r > h_n > h_s \Rightarrow w_n*k*r > h_s \Rightarrow w_n*k > h_s/r \Rightarrow w_n*k > w_s$

Corollary We assume that the set of *NODES* for every level of an R-Tree satisfies the conditions of Theorem 1 for a search region. In addition we assume that the R-Tree contains *log(n)* levels (balanced tree with *n* points) and the search region contains *m* of these points. In this case the asymptotic search complexity is ***O(log(n)∗(m+6∗k+28))***.

III TREE PACKING METHOD

Existing tree packing methods do either not consider the ratio for leaf nodes (e.g. [1], [17], [19] and [23]) or take to much time to generate the R-Tree (e.g. [13]). Therefore this section introduces a new tree packing method. This method will ensure that no nodes strongly overlap. The method has been also designed to be fast, facilitating frequent regenerations of the R-Tree after a certain number of entries are added to the database.

The allowed degree of deviation from the form of envisaged search conditions is given by factor *k*. Note that this factor is not an input for the method. Instead a higher value for *k* allows larger deviations from the search ratio making it easier to meet the specified conditions. The price for a larger deviation is a weaker upper bound for the number of visited nodes.

The proposed tree packing method has two phases. During a first phase a “tree plan” will be generated. This tree plan defines the full tree structure with all nodes and the number of entries in every subtree. The tree plan is generated purely from the number *n* of entries (not from the entries itself!), the number of successors *w* of an inner node and the capacity *v* of leaf nodes. This tree plan ensures that the tree is balanced and contains only the minimum number of nodes.

During the second phase the entries are distributed top down from the root to the leaf nodes. The core of this method is a procedure for a binary split. This splitting procedure chooses the dimension for the split according to the ratio parameter. The idea is to choose a dimension for which the bounds of the two resulting sets of points are closest to the ratio. For this purpose the length of “edges” for point sets are divided by the corresponding ratios. The dimension with the greatest quotient is chosen for the split to reduce the deviation from the ratio.

The starting point for a tree plan is the computation of the number of nodes for every level *i* of the tree by the function *level(i)*. Here *level(0)* refers to leaf nodes in the tree, *level(1)* to parents of the leaf nodes, etc. It is probably obvious that the minimum number of leaf nodes with capacity *v* needed for *n* elements is given as follows:

$$level(0) = \lceil n / v \rceil$$

If we have at maximum *w* successors for every node then we need to have the following number of parent nodes on level *i+1* for the child nodes on level *i*:

$$level(i+1) = \lceil level(i) / w \rceil$$

Now we will distribute the *n* entries as equal as possible:

- *n mod level(0)* leaf nodes with $\lceil n/level(0) \rceil$ elements
- *level*(*0*) - n mod *level*(*0*) leaf nodes with $\lfloor n/level(0) \rfloor$ elements

With this approach the differences between the numbers of entries of leaf nodes are at maximum one. In a similar way we assign *level(i)* child nodes to *level(i+1)* parent nodes:

- *level*(*i*) mod *level*(*i+1*) inner nodes with $\lceil level(i)/level(i+1) \rceil$ elements
- *level*(*i+1*) - *level*(*i*) mod *level*(*i+1*) inner nodes with $\lfloor level(i)/level(i+1) \rfloor$ elements

During the last step we label the nodes of the tree bottom up with the number of elements in their corresponding subtree. For this purpose every leaf node is labeled with the number of contained elements and every inner node is labeled with the sum of all labels from its children.

The second phase of the tree generation procedure recursively splits the set of entries into subsets according to the tree plan. For this purpose the procedure assigns the full set of entries to the root node. Then the procedure performs the following steps for every node starting with the root node:

- Choose an inner node to which a subset of entries was already assigned.
- Split the set related to the selected node into *w* subsets with $n_1, \ldots, n_w$ elements if *w* is the number of children and $n_1, \ldots, n_w$ are the labels for the child nodes.

In this paper we propose a method which is based on a sequence of binary splits. A binary split generates two sets by dividing a set of entries into two subsets according to their values of a selected dimension. This means that *w-1* binary splits are needed for distributing the set of entries to *w* child nodes. Therefore the binary splitting continues until every generated set refers exactly to one child node.

For a single binary split the dimension with the maximum deviation from the predefined ratio will be chosen. Using these ideas we introduce the procedures "Split" and "BinSplit" (“binary split”).

Method Split

```
Input:        set S of n entries,
              w sizes [n1,..., nw]
              ratio   [r1,. . ., rd]
Output:       list sets with specified size
if (w == 1) return [ S ]
Split sizes [n1,...,nw] into two separate sequences:
  sizes1:=[n1,...,nv]
  sizes2:=[nv+1,...,nw]
[s1,s2]:=BinSplit(S,n1+...+nv,nv+1+...+nw,
                  [r1,..., rd])
return
  Split(s1,[n1,...,nv],[r1,...,rd])
  append Split(s2,[nv+1,...,nw],[r1,...,rd])
```

Method BinSplit

```
Input:        set S,
              size1 and size2
              ratio [r1,...,rd]
Output:       set1 with size1
set2 with size2
Identify lower and upper bounds for the set S:
  [l1,. . . , ld]
```

```
  [u1,. . . , ud]
Identify dimension with greatest deviation from ra-
tio:
  iMax = 1
  max = (u1-l1)/r1
  for every dimension i>1:
  if max < (ui-li)/ri then
    iMax = i
    max = (ui-li)/ri
Compare according to dimension iMax:
  Assign size1 smallest elements to set1
  Assign remaining size2 entries to set2
```

IV First experiments

This section summarizes first results from experiments with the new tree packing method. These experiments serve the purpose to analyze the following two questions:

- Is it possible to generate an R-Tree according to our requirements?
- How does the search time grow for an R-Tree produced by the proposed tree packing method?

In particular the second point requires a careful analysis of the relationship between the total number of entries, the size of the result set and the search time excluding all other parameters. For this purpose we use synthetic data providing more control over parameters like size of a data set and distribution of entries. Since equally distributed data does not provide a real challenge for tree packing methods we use a Gaussian distribution in two dimensions. This distribution makes it easy to analyze weaknesses of tree packing algorithms since the density of data smoothly changes across the value space.

In addition we want to avoid influences of other factors as for example task switching, page faults or CPU usage by other tasks. Therefore our implementation returns the number of visited nodes instead of the elapsed time for a search. We assume that the search time is dominated by the time needed for loading a block from secondary memory. We also assume that a database provides a cache in main memory containing frequently used disk blocks. In our case this cache will most probably contain blocks representing inner nodes but probably (almost) no leaf nodes because we have far more leaf nodes than inner nodes. Therefore we will just count leaf nodes and no inner nodes. According to our theorem we expect a **constant** quotient *m/t* (constant access time) which is independent from the size of the database (number of entries *n*).

Also the search conditions need to be chosen carefully to ensure equal conditions for data sets with different sizes. We have chosen a workload containing search conditions along the vertical center line in the normal distributed data set. The size of a search condition in both dimensions is half the size of the condition for the next larger data set containing four times more entries. This means that the conditions retrieve roughly the same number of entries for the next larger data set. We also double the number of search conditions for the next larger data set to ensure that they cover the same space on this center line.

For all tests we compare the new tree packing methods (identified by *Bounds*) with two other well known methods to check differences and commonalities. The first method is the Sort Tile Recursive method [19] (*STR*) which is probably the best bottom up method. The second method is the Priority Tree [1] (*PRIO*) as one top down method. The Priority Tree ensures a time complexity of $O(n^{(d-1)/d})$ which is the best we can achieve for a non-replicating index structures in general.

All experiments need to consider the ratio between the height and the width of a search region. Traditional tree packing methods try to generate nodes which are close to a square. Therefore we include search conditions with equal height and width (ratio = 1). In addition we investigate the case where we have a ratio of 10 between width and height.

We need to choose the factor *k* from the precondition of our theorem as well. A high value for *k* allows larger deviations from the search ratio. This makes it easier for the tree packing to meet the precondition of this theorem. The "price" is however a weaker upper bound for the number of visited nodes.

With a ratio $r = h_s / w_s = 1$ and a factor $k=10$ we get the following bounds for the height h_n and width w_n of nodes:

$$w_n * 1/10 < h_n < w_n * 10 * 1 \Leftrightarrow \mathbf{w_n / 10 < h_n < w_n * 10}$$

For the ratio $r = h_s / w_s = 1/10$ we get the following condition:

$$w_n * 1/10/10 < h_n < w_n * 10 * 1/10 \Leftrightarrow \mathbf{w_n / 100 < h_n < w_n}$$

All trees are generated with 20 successors for inner nodes and 20 points for leaf nodes. We consider 6 data sets from 1024 (2^{10}) entries up to 1,048,576 (2^{20}) entries.

We investigate the Search Tile Recursive method first. The following table summarizes the results. The first column contains the number of entries. Second and third column contain the percentage of nodes **not** satisfying the form condition for the ratio 1:1 and 1:10. Please note that we allow a deviation of k = 10, meaning that these "bad" nodes are very different from the required form. The last four columns provide the quotient between the number of search results and the search time (equals number of leaf nodes in out setup) in the averages case and in the worst case (labeled by min) with our workload. The numbers in brackets provide the number of visited leaf nodes if the "worst case search" did not provide any search results.

STR	Form %		Time 1:1		Time 1:10	
	1:1	1:10	aver	min	aver	min
1,024	0.00	61.54	9.34	3.75	5.48	0.60
4,096	1.95	56.59	7.94	1.33	5.04	*(2)*
16,384	6.10	51.59	10.15	0.67	5.31	*(6)*
65,536	5.00	53.68	7.33	0.33	4.85	*(7)*
262,144	4.84	52.56	7.96	*(3)*	4.71	*(10)*
1,048,576	4.60	52.14	6.63	*(4)*	4.39	*(10)*

The performance of STR is not too bad. In the case of quadratic searches most nodes satisfy the form condition. In the case of a search with a ratio of 1:10 we have however more than half "bad" nodes, because STR does not consider the ratio parameter. The average search time slightly increases (note that we consider *m/t* in this table meaning we have decreasing values). In the worst case we have a much stronger increase of search time for both types of search conditions.

The next table summarizes the result for the Priority Tree.

PRIO	Form %		Nodes 1:1		Nodes 1:10	
	1:1	1:10	aver	min	aver	min
1,024	11.67	40.00	6.78	4.71	5.81	3.00
4,096	20.63	40.08	5.74	*(1)*	5.01	*(1)*
16,384	26.37	43.04	7.19	*(1)*	4.01	*(1)*
65,536	28.35	44.35	5.50	*(1)*	3.37	*(2)*
262,144	30.02	44.26	6.80	*(2)*	4.34	*(2)*
1,048,576	31,01	44.19	5.41	*(2)*	2.83	*(9)*

The overall performance of the Priority Tree seems to be worth than for STR. This is true for the form of nodes as well as for the average search time. The Priority Tree seems to have a small advantage for the worst case queries because fewer nodes need to be searched if a search does not return any results.

The final table provides the results for the new method.

Bounds	Form %		Nodes 1:1		Nodes 1:10	
	1:1	1:10	aver	min	aver	min
1,024	0.00	0.00	10.84	5.00	11.49	3.00
4,096	0.49	0.98	8.29	*(1)*	9.24	*(1)*
16,384	0.24	0.24	10.0	*(0)*	8.50	*(1)*
65,536	0.06	0.03	9.43	*(1)*	9.14	*(2)*
262,144	0.00	0.02	8.35	*(1)*	8.56	*(2)*
1,048,576	0.00	0.01	8.25	*(2)*	8.50	*(3)*

The table shows that this method easily outperforms the two other methods for larger number of entries. This is true for the form of nodes, the average search time and the worst case. Although the search time is not really constant there is also no steady increase in search time (average and worst case).

5. SUMMARY

The key problem for database applications using an R-Tree is the lack of knowledge how well this index structure supports searches. This means that database applications take the risk that search times grow significantly if the amount of data increases or even just changes. This paper addresses this issue by providing a criterion under which efficient searching is guaranteed. Even before this paper it was probably obvious that incompatibilities between forms of nodes and search regions are the cause for inefficient searches. Now this paper could prove that for two dimensional R-Trees compatibility ensures logarithmic search times. With this result the paper nicely marks the borderline between logarithmic and non-logarithmic search time complexity for two dimensional R-Trees.

The result provides also key requirements for the design of future R-Tree algorithms. As our tests already show, existing algorithms often fail to generate nodes with the "right" form On the other hand the new tree packing method clearly showed the benefit of using ratio information. An experimental database ([21], [3]) was also implemented using this method. The database could demonstrate the applicability and efficiency of this approach by a sample application managing a gazetteer with more than 7 million place names.

REFERENCES

[1] Lars Arge, Mark de Berg, Herman J. Haverkort, Ke Yi: The Priority R-tree: "A Practically Efficient and Worst-Case Optimal R-tree", Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, p. 347 - 358, Paris, France,2004

[2] C. Ang, T. Tan: "New linear node splitting algorithm for R-Trees" In M. Scholl and A. Voisard (Eds.), Advances in Spatial Databases, LNCS, Springer-Verlag 1997

[3] A. Bär: "Implementierung, Optimierung und Vergleich ausgewählter Tree-Packing-Verfahren für den R-Baum im Rahmen einer räumlichen Datenbank", Diplomarbeit, Fachhochschule Hof, 2005

[4] J. L. Bentley: "Multidimensional Binary Search Trees Used for Associative Searching" Communication of the ACM, 18(9), September 1975

[5] J. L. Bentley: "Multidimensional divide and conquer" Communication of the ACM, 23(6), pages 214 - 229, 1980

[6] J. L. Bentley, H. A. Maurer: "Efficient worst-case data structure for range searching" Acta Informatica, 1980

[7] S. Berchtold, D. A. Keim, H. P. Kriegel: "The X-Tree: An Index Structure for High-Dimensional Data", Proc. International Conference on Very Large Data Bases, pages 28 – 39, Mumbai (Bombay), India, 1996

[8] N. Beckmann, H. P. Kriegel, R. Schneider, B. Seeger: "The R*-tree: An Efficient and Robust Access Method for Points and Rectangles", Proc. ACM-SIGMOD International Conference on Management of Data, pages 322 - 331,„Atlantic City (NY), May 1990

[9] R. Bayer, C. McCreight: "Organization and Maintenance of Large Ordered Indexes", Acta Informatica 1(3), pages 173 - 189, 1972

[10] S. Brakatsoulas, D. Pfoser, Yannis Theodoridis: "Revisiting R-Tree Construction Principles", ABDIS 2002, Springer LNCS 2435, Bratislava, Slovakia, 2002

[11] R. Göbel, A. Almer, T. Blaschke, G. Lemoine, A. Wimmer: "Towards an Integrated Concept for Geographical Information Systems in Disaster Management", Proceedings First International Symposium on Disaster Management, Delft, Springer 2005

[12] V. Gaede, O.Günther: "Multidimensional Access Methods", Interner Bericht (www.wiwi.fu-berlin.de), Humboldt Universität, Berlin, 1997

[13] Y. J. Garcia, M. A. López, S. T. Leutenegger: "A greedy algorithm for bulk loading R-Trees", Proc. 6th ACM Symposium on Advances in GIS, pages 163 – 164, 1998

[14] R.Göbel: „Effiziente Verwaltung geographischer Daten mit räumlichen Datenbanksystemen – Möglichkeiten und Grenzen", Tagungsband AGIT 2005, Salzburg, Wichmann Verlag, 2005

[15] A. Guttman: "R-Trees: A Dynamic Index Structure for Spatial Searching", Proc. ACM SIGMOD Conference, Boston, pages 47 - 57, 1984

[16] J. M. Hellerstein, E. Koutsoupias, C. H. Papadimitriou: "On the Analysis of Indexing Schemes. ", Proc. ACM Symposium on Principles of Database Systems, Tucson (Arizona), May 1997

[17] I. Kamel, C. Faloutsos: "On Packing R-Trees" Proc. 2nd International Conference on Information and Knowledge Management (CKIM-93), pages 490 - 499, Arlington, 1993

[18] K.V.R. Kanth, A.K. Singh: "Optimal Dynamic Range Searching in Non-replicating Index Structures", Proc. International Conference on Database Theory, LNCS 1540, pages 257-276, 1999

[19] S. T. Leutenegger, J. Edgington, M. A. Lopez: "STR: A Simple and Efficient Algorithm for R-Tree Packing", Proc. 12 th International Conference on Data Engineering, pages 497 - 506, 1997

[20] K. Mehlhorn: "Data Structures and Algorithms 3: Multidimensional Searching and Computational Geometry", Springer Verlag 1984

[21] S. Neupert: "Entwurf und Realisierung einer räunlichen Datenbank auf der Basis optimierter R-Bäume", Diplomarbeit, Fachhochschule Hof, 2005

[22] K.V. Ravi Kanth, A. Singh: "Optimal Dynamic Range Searching in Non-Replicating Index Structures" Proc. International Conference on Database Theory, pages 257 - 276, Springer Verlag Berlin Heidelberg, 1999

[23] N. Roussopoulos, D. Leifker: "Direct Spatial Search on Pictorial Databases Using Packed R-Trees", Proc. ACM SIGMOD, 1985

[24] T. Sellis, N. Roussopoulos, C. Faloutsos: "The R+-Tree: A Dynamic Index for Multi-Dimensional Objects", Proc. International Conference on Very Large Data Bases, pages 507 – 518, Brighton, September 1987

[25] Y. Theodoridis, T. Sellis: "A Model for the Prediction of R-Tree Performance", Proceedings of the 5th ACM Symposium on Principles of Database Systems, pages 161-171, 1996

A Novel Approach For Mining Emerging Patterns in Rare-class Datasets

Hamad Alhammady
Etisalat University College, UAE
Email: hamad@euc.ac.ae

Abstract - Mining emerging patterns (EPs) in rare-class databases is one of the new and difficult problems in knowledge discovery in databases (KDD). The main challenge in this task is the limited number of rare-class instances. This scarcity limits the number of emerging patterns that can be mined for the rare class. In this paper, we propose a novel approach for mining emerging patterns in rare-class datasets. We experimentally prove that our method is capable of gaining enough knowledge from the rare class; hence, it increases the performance of EP-based classifiers.

I. INTRODUCTION

The rare-class problem is faced in many real life applications. These applications include direct marketing, web log analysis, and intrusion detection in security networking systems. The main challenge in this problem is the scarcity of the rare cases. This challenge prevents classification methods from gaining enough knowledge from the rare class.

In this paper, we introduce a new method for mining emerging patterns (EPs) in rare-class datasets. EPs are a new kind of patterns introduced recently [2]. They have been proved to have a great impact in many applications [1] [5] [6] [8] [9]. EPs can capture significant changes between datasets. They are defined as itemsets whose supports increase significantly from one class to another. The discriminating power of EPs can be measured by their growth rates. The growth rate of an EP is the ratio of its support in a certain class over that in another class. Usually the discriminating power of an EP is proportional to its growth rate.

For example, the Mushroom dataset, from the UCI Machine Learning Repository [7], contains a large number of EPs between the poisonous and the edible mushroom classes. Table 1 shows two examples of these EPs. These two EPs consist of 3 items. e1 is an EP from the poisonous mushroom class to the edible mushroom class. It never exists in the poisonous mushroom class, and exists in 63.9% of the instances in the edible mushroom class; hence, its growth rate is ∞ (63.9 / 0). It has a very high predictive power to contrast edible mushrooms against poisonous mushrooms. On the other hand, e2 is an EP from the edible mushroom class to the poisonous mushroom class. It exists in 3.8% of the instances in the edible mushroom class, and in 81.4% of the instances in the poisonous mushroom class; hence, its growth rate is 21.4 (81.4 / 3.8). It has a high predictive power to contrast poisonous mushrooms against edible mushrooms.

TABLE I
EXAMPLES OF EMERGING PATTERNS

EP	Support in poisonous mushrooms	Support in edible mushrooms	Growth rate
e1	0%	63.9%	∞
e2	81.4%	3.8%	21.4

e1 = {(ODOR = none), (GILL_SIZE = broad), (RING_NUMBER = one)}
e2 = {(BRUISES = no), (GILL_SPACING = close), (VEIL_COLOR = white)}

II. RELATED WORK

Traditional classification accuracy (percentage of correctly classified instances in all classes) is not a suitable metric to measure the performance of classifiers in the rare-class problem. For example, suppose we have an imbalanced dataset consisting of two classes, and the major class contributes 95% of the instances. Then, the traditional accuracy can be increased to at least 95% by assuming all data instances belong to the major class.

TABLE II
CONFUSION MATRIX

	Classified as major class	Classified as rare class
Actual Major class	*TM*	*FR*
Actual Rare class	*FM*	*TR*

TM = number of major-class instances classified as major-class instances
FR = number of major-class instances classified as rare-class instances
FM = number of rare-class instances classified as major-class instances
TR = number of rare-class instances classified as rare-class instances

According to the confusion matrix in table 2, the traditional accuracy is defined as follows.

$$Accuracy = \frac{TM + TR}{TM + FR + FM + TR} \quad (1)$$

This accuracy is dominated mainly by the performance of classifiers on the major class. This is because of the large ratio between the number of major-class instances and the number of rare-class instances in the training set. The *F-measure* (F) [10] is a suitable alternative metric to evaluate

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 207–211.

classifiers in rare-class classification. This metric evaluates a classifier based on both *precision* (P) and *recall* (R) as follows.

$$P = \frac{TR}{TR + FR} \tag{2}$$

$$R = \frac{TR}{TR + FM} \tag{3}$$

$$F = \frac{2PR}{P + R} \tag{4}$$

The F-measure has been used to evaluate a number of classification methods designed specifically for rare-class problems.

There is a number of techniques proposed for rare-class problems. On of these techniques is EPRC [3]. This approach is based on applying some improving stages to maximize the discriminating power of rare-class EPs. These stages include generating new undiscovered rare-class EPs, pruning low-support EPs, and increasing the support of rare-class EPs.

EPDT [4] aims at supporting decision trees in rare-class problems. It consists of two steps. First, new non-existing rare-class instances are generated. Second, the most important rare-class instances are over sampled. These two steps increase the performance of decision trees as they work together toward balancing the rare class with the major class.

Two-phase rule induction (PNrule) [11] tries to find the best tradeoff between recall and precision to achieve the highest possible f-measure. It consists of two phases. In the first phase it seeks high recall objective using P-rules. These P-rules detect the presence of the target class. In the second phase the technique seeks high precision objective using N-rules. These N-rules detect the absence of the target class. The P-rules, mined in the first phase, are not accurate. The reason is that they cover many major-class instances beside the rare-class instances. This is because the high interference between the major and rare classes due the scarcity of the rare-class. This problem affects the f-measure negatively.

III. EMERGING PATTERNS AND CLASSIFICATION

Let $obj = \{a_1, a_2, a_3, \ldots a_n\}$ is a data object following the schema $\{A_1, A_2, A_3, \ldots A_n\}$. $A_1, A_2, A_3 \ldots. A_n$ are called attributes, and $a_1, a_2, a_3, \ldots a_n$ are values related to these attributes. We call each pair (attribute, value) an item.

Let I denote the set of all items in an encoding dataset D. *Itemsets* are subsets of I. We say an instance Y contains an itemset X, if $X \subseteq Y$.

Definition 1. Given a dataset D, and an itemset X, the support of X in D, $s_D(X)$, is defined as

$$s_D(X) = \frac{count_D(X)}{|D|} \tag{5}$$

where $count_D(X)$ is the number of instances in D containing X.

Definition 2. Given two different classes of datasets D_1 and D_2. Let $s_i(X)$ denote the support of the itemset X in the dataset D_i. The growth rate of an itemset X from D_1 to D_2, $gr_{D_1 \to D_2}(X)$, is defined as

$$gr_{D_1 \to D_2}(X) = \begin{cases} 0, & \text{if } s_1(X) = 0 \text{ and } s_2(X) = 0 \\ \infty, & \text{if } s_1(X) = 0 \text{ and } s_2(X) \neq 0 \\ \frac{s_2(X)}{s_1(X)}, & \text{otherwise} \end{cases} \tag{6}$$

Definition 3. Given a growth rate threshold $\rho > 1$, an itemset X is said to be a ρ-emerging pattern (ρ-EP or simply EP) from D_1 to D_2 if $gr_{D_1 \to D_2}(X) \geq \rho$.

Let $C = \{c_1, \ldots c_k\}$ is a set of *class labels*. A *training dataset* is a set of data objects such that, for each object *obj*, there exists a class label $c_{obj} \in C$ associated with it. A *classifier* is a function from attributes $\{A_1, A_2, A_3, \ldots A_n\}$ to class labels $\{c_1, \ldots c_k\}$, that assigns class labels to unseen examples.

IV. MINING EPS IN RARE-CLASS DATASETS

The major problem in mining EPs in rare-class datasets is that the number of the rare-class EPs is very small compared to the major-class EPs. Work in [3] aims at solving this problem by generating additional EPs. In this paper, we propose a novel method for mining a large number of rare-class EPs to fill the gap between the rare class and the major class.

First, let us investigate the main reason behind the shortage in rare-class EPs. Mining EPs involves some sort of comparison between the small population in the rare class and the large population in the major class. That is, the mining process aims at finding patterns that exist frequently in the small number of rare-class instances and that do not exist very frequently in the large number of major-class instances. This difficult restriction limits the number of rare-class EPs because the rare-class instances are compared at the same time with all the major-class instances.

Our proposed approach is based on mining rare-class EPs by comparing the rare-class instances with subsets of the major-class instances instead of the whole range of data.

The details of our approach are as follows. Suppose that the rare class (*RC*) and the major class (*MC*) consist of *R* and *M* instances, respectively. The major class is divided into a number of subsets, MS_j such as $j = \{1, \ldots., M/R\}$ and the number of instances in each subset is *R*. Rare-class EPs are mined from *RC* against each subset of *MC*. That is the mining process is divided into *M/R* sub processes rather than one as in the normal case. The results of each sub process are a reasonable number of rare-class EPs because the number of instances in RC and each subset MS_j is identical. These rare-class EPs are distributed in M/R sets ($REPS_j$) each of which is related to one of the mining sub processes.

The rare-class EPs in the M/R sets are combined in one set, the EPs are then ranked in a descendent order according to their strength. The strength of an EP e, $strg(e)$, is defined as follows.

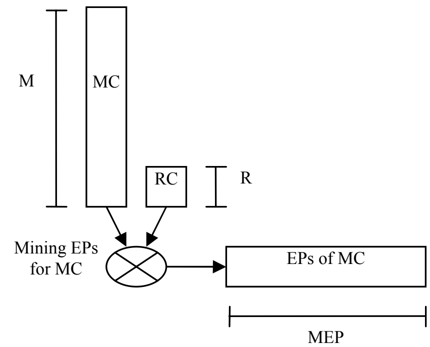
$$strg(e) = \frac{gr(e)}{1+gr(e)} * s(e) \tag{7}$$

The strength of an EP is proportional to both its growth rate (discriminating power) and support. Notice that if an EP has a high growth rate and a low support its strength might be low. In addition, if it has a low growth rate and a high support its strength might also be low.

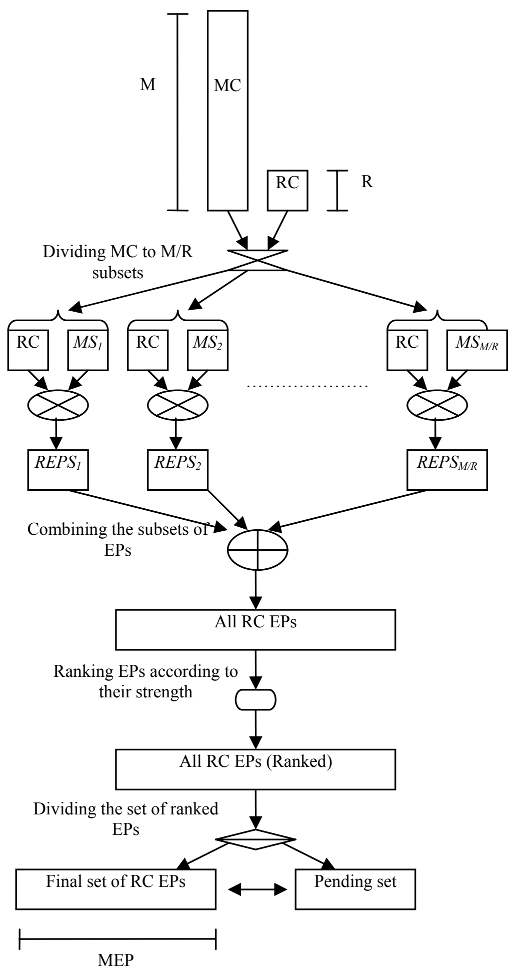
Suppose that the number of major-class EPs is MEP. The set of ranked rare-class EPs is divided into two subsets. The first subset (called the final subset) contains MEP strongest rare-class EPs. That is, the number of final rare-class EPs equals the number of major-class EPs. The second subset (called the pending subset) contains the remaining rare-class EPs that are not yet included in the final subset.

The final stage of our approach involves comparing the final subset of rare-class EPs with the major-class EPs. If an EP exists in both the final subset of rare-class EPs and the major-class EPs, then it is eliminated from the final subset and the strongest EP in the pending subset is added to the final subset. This process ensures that noisy rare-class EPs are eliminated from the final subset and the strongest EPs are added to this subset.

- After completing our approach, we end up with two sets of EPs. On of them is related to the major class and the other is related to the rare class. These two sets have almost the same number of EPs. That is, rare-class EPs are not rare compared to the major-class EPs. The two sets can be used with any EP-based classifier (such as BCEP [5]) to classify unlabeled instances in both classes. Figure 1 sketches our proposed method.



(a)



(b)

Figure1. (a) Mining major-class EPs. (b) Mining rare-class EPs using DEP.

Rare-class EPs mining using our method (we call it division for mining EPs - DEP) are motivated by the following points:

- Dividing the major class into subsets enables the discovery of unseen rare-class EPs. These unseen EPs are covered by the overwhelming amount of data in the major class.
- Using the strength function to evaluate the rare-class EPs ensures that noisy EPs have minimum effect.

V. EXPERIMENTAL EVALUATION

We conduct experiments on 12 datasets from UCI repository of machine learning databases [7]. These datasets are disease, hypothyroid, sick-euthyroid, and nine binary datasets formed from the king-rook-king dataset[1]. Table 3 lists these datasets and the percentage of the rare-class for each one of them. We compare our proposed method (DEP) with Pnrule [11], boosted PNrule, EPRC [3], and EPDT [4]. The results are shown in table 4[2].

The following points summarize the results:

- Our proposed method, DEP, outperforms all the other methods on all datasets.
- DEP has the highest average F-measure.

VI. CONCLUSIONS

Mining emerging patterns (EPs) in rare-class datasets is one of the challenging problems in data mining. This problem is considered as the main reason behind the failure of EP-based classifiers in the rare-class classification. In this paper, we propose a new technique for mining EPs in rare-class datasets. Our proposal is based on dividing the mining process into a number of sub processes and then combining the resulted EPs according to their strength. We experimentally prove that our method is effective in rare-class classification.

TABLE III
RARE-CLASS DATABASES

Dataset	Percentage of the rare class
Disease	1.6
Hypothyroid	4.7
Sick-euthyroid	9.2
KRK-5	1.7
KRK-8	5.1
KRK-9	6.1
KRK-10	7.1
KRK-11	10.2
KRK-13	14.9
KRK-14	16.2
KRK-15	7.7
KRK-16	1.4

TABLE IV
F-MEASURE COMPARATIVE RESULTS

Datasets	PNrule	BPNrule	EPRC	EPDT	DEP
Disease	-	-	73.5	74.9	**76.8**
H-thyroid	-	-	93.7	94.3	**95.6**
S-thyroid	-	-	88.3	88.7	**90.2**
KRK-5	63.5	65.8	65.1	65.5	**68.4**
KRK-8	52.7	61.8	66.1	66.9	**69.8**
KRK-9	43.4	59.1	66	66.7	**70.1**
KRK-10	42.1	54.6	58.2	55.9	**63.3**
KRK-11	49	58.6	58.9	58.3	**64.9**
KRK-13	58.5	61.6	64.5	65.3	**69.5**
KRK-14	61.7	72.9	74.3	74	**78.8**
KRK-15	66.1	72.1	74.9	74.8	**77.4**
KRK-16	56.4	70.2	78.5	78.2	**83.3**
Average	54.8	64.1	67.4	67.3	**71.7**

REFERENCES

[1] H. Alhammady, and K. Ramamohanarao. Expanding the Training Data Space Using Emerging Patterns and Genetic Methods. In Proceeding of the 2005 SIAM International Conference on Data Mining, New Port Beach, CA, USA.
[2] G. Dong, and J. Li. Efficient Mining of Emerging Patterns: Discovering Trends and Differences. In Proceedings of the 1999 International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA.
[3] H. Alhammady, and K. Ramamohanarao. The Application of Emerging Patterns for Improving the Quality of Rare-class Classification. In Proceedings of the 2004 Pacific-Asia Conference on Knowledge Discovery and Data Mining, Sydney, Australia.
[4] H. Alhammady, and K. Ramamohanarao. Using Emerging Patterns and Decision Trees in Rare-class Classification. In Proceedings of the 2004 IEEE International Conference on Data Mining, Brighton, UK.
[5] H. Fan, and K. Ramamohanarao. A Bayesian Approach to Use Emerging Patterns for Classification. In Proceedings of the 14th Australasian Database Conference (ADC'03), Adelaide, Australia.
[6] Guozhu D., Xiuzhen Z., Limsoon W., and Jinyan L.. CAEP: Classification by Aggregating Emerging Patterns. In Proceedings of the 2nd International Conference on Discovery Science (DS'99), Tokyo, Japan.
[7] C. Blake, E. Keogh, and C. J. Merz. UCI repository of machine learning databases. Department of Information and Computer Science, University of California at Irvine, CA, 1999.
http://www.ics.uci.edu/~mlearn/MLRepository.html.
[8] H. Alhammady & K. Ramamohanarao. Mining Emerging Patterns and Classification in Data Streams. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI), Compiegne, France (Sep 2005), pp. 272-275.
[9] H. Alhammady & K. Ramamohanarao. Using Emerging Patterns to Construct Weighted Decision Trees. In IEEE

[1] A binary rare-class problem is formed by considering a class as a rare class and union of the other classes as one major class.
[2] The first three datasets are not included in the average due to the unavailable results for PNrule. We could not find more results for this technique from published research neither from their authors.

This page intentionally blank

Transactions on Knowledge and Data Engineering. Volume 18, Issue 7 (July 2006), pp. 865-876.

[10] Van Rijsbergan, C. J. (1979). Information retrieval. London, UK: Butterworths.

[11] Joshi, M. V., Agarwal, R. C., & Kumar, V. (2001). Mining needle in a haystack: classifying rare classes via two-phase rule induction. In Proceedings of the ACM-SIGMOD International Conference on Management of Data (ACM SIGMOD), Santa Barbara, CA, USA, pp. 91-102.

Integration of Flexible Manufacturing and Change Management Processes in a Service-Oriented Architecture

Hardy Krappe, Stilian Stanev and Jivka Ovtcharova

University of Karlsruhe, Institute for Information Management in Engineering (IMI), Karlsruhe, Baden-Württemberg, 76131, Germany

ABSTRACT — The success of a company invariably depends more and more on the ability of a company to recognize changes in its environment at an early stage and consider these changes timely in its competitive behavior. Flexibility, as one of the major factors, marks the ability of a company to master complex environmental situations in order to boost its chances to survive and safeguard its long-term success. This article reflects on the perspectives offered by the utilization of the service oriented paradigm for realizing the integration of a flexibility evaluation platform used for 'near real-time' flexibility measurement and monitoring of a manufacturing system, within the scope of a change management platform.

***Index Terms* — Flexibility, SOA, Web Services, Production System, Business Processes**

I. Introduction

Since the last decade, producing companies are confronted with an ever more complex and fast-changing environment, which is the result of an increasing individualization of the customers. They are forced to integrate a rising number of products and product varieties whereas the predictability for manufacturing determinations like break evens, turnovers or sales is decreasing [1]. Accordingly, the success of a company depends more and more on the ability to recognize internal and external influences as early as possible and to react to resulting changes with an adaptation of their organizational structures, and eventually their production systems.

To ensure a fast response to these influences the most critical challenge for manufacturing companies today is the management of product and production changes in general, not only within a production system of one factory but also across organizational borders. That implies a permanent alertness of manufacturing companies regarding for instance the flexibility of their production systems. At present, most major manufacturing companies and OEMs[1] consider flexibility as an important performance factor representing a significant feature in order to master the complex situations on the market and at the same time to ensure a long-run success of companies [2].

However, since as stated in [19], flexibility cannot be properly considered in the decision making process if it is not properly defined in a quantitative fashion, quantified data has to be gathered from a variety of sources and in a near real-time frequency. Another viewpoint is that flexibility is a relative attribute that depends not only on the manufacturing system itself, but also on the external demands placed upon it [18].

To enable a long-run success a serious need to adapt existing business processes according to the demands from customers, suppliers or laws has to be realized. Whereas the approaches from the past focused on static applications and legacy systems, applications operating in modern dynamic changing environments need a suitable IT-Architecture. Traditional IT-Systems are inadequately architected to meet the rapidly evolving needs of scalable, platform independent enterprise applications [8] [9].

The Service-Oriented Architecture (SOA) is a new software development paradigm developed to support dynamic changing enterprise applications and open, flexible, agile software systems.

An added advantage of SOA is that applications can be composed at runtime using already existing services.

II. Perspectives

Funded by the European Union[2] the X-Change project aims to develop and integrate flexibility measurement methods to support and improve change management processes within production systems. The project will provide a software framework that will extend and enhance the effectiveness of production systems in general by optimizing change processes within a production network.

[1] Original Equipment Manufacturers

[2] FP6-2004-IST-NMP-2

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 213–218.

To achieve this objective three general topics have to be addressed [15]:

1) The development of a platform for measuring the flexibility of a production enterprise by the means of suitable flexibility evaluation methods.
2) The integration of the flexibility evaluation platform in a flexible and agile service oriented framework which facilitates change management in production systems and extended enterprises which operate in dynamically changing environments.
3) The provision of decision-support through the software framework regarding the management and planning of short and mid-term changes of a manufacturing system as well as a continuous redesign and improvement of change management processes in the product lifecycle.

The X-Change vision is illustrated in Figure 1.

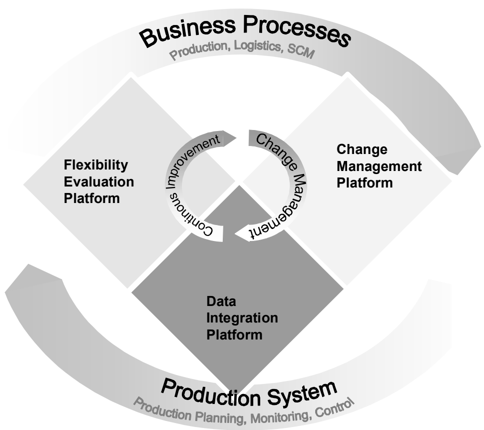


Figure 1 Continuous improvement of change processes within the product lifecycle

It elucidates the interwoven processes of continues improvement and change management within the business processes of production systems. The concept presented in Figure 1 connects formerly separated domains like manufacturing systems (products) and change management (processes) with each other. Thus it enables a continuous improvement both of processes and of next generation products by integrating them in a service-oriented architecture.

In order to assemble the project objectives the following five subtasks have to be accomplished:

1) Development of a Data Integration Platform
2) Integration of existing business process of the involved companies in DIP
3) Integration of a change management platform
4) Integration of the flexibility evaluation platform on IT level
5) Provision of an easy-to-use human-machine interface

The focus of this paper is the integration of the flexibility measurement platform together with a change management platform into a service-oriented architecture to achieve the maximum flexibility of the applicable software systems.

III. Service-oriented Architecture in a nutshell

The following section provides an outline to the essential advantages and characteristics of SOAs.

Literature offers several definitions of service-oriented architecture [4] [16] [17]. Regarding the context of flexible and agile production systems the authors consider SOAs as an evolution of past middleware platforms. They are preserving successful characteristics of traditional architectures, and bringing with it, distinct principles that foster service-orientation in support of a service-oriented enterprise (SOE). Contemporary SOAs represent an open, agile, extensible, federated, composable architecture compromised of autonomous, quality of service capable, vendor diverse, interoperable discoverable and potentially reusable services [4]. In essence SOAs can be characterized by a number of discrete, organized services for an end-to-end solution. Typically these services come on two flavours:

1) Business Services containing implemented business logic (processes and rules) and
2) Technical Services supporting technical functionality required to ensure the smooth operation of the overall solution. This might include data services (for persisting business objects), services for authentication or identification, or even services that provide online access to catalogues of other services.

This approach is very similar to the component based architectures of the late 90s but the main difference is that SOA takes a more coarse-grained view of functionality.

In general SOAs persist of three basic attributes: *autonomous, interoperable* and *composable*.

Autonomous services means that the data inside the SOA is private to the service and always encapsulated by the service so the only way of accessing it is through the business logic of the service. This data is only loosely correlated to the data on the outside traveling in form of messages. Furthermore *autonomous* services can be distinguished into three principle properties [7].

1) They are created independently of each other.
2) They operate free in their environment.
3) They provide self contained functionality.

Interoperability means that SOAs facilitate standards that are based on well-known protocols and description or data exchange languages such as XML, WSDL, UDDI, SOAP, BPEL, HTTP, CPP, ebXML, ebSOA, FERA, OWL-S and WS-BPEL.

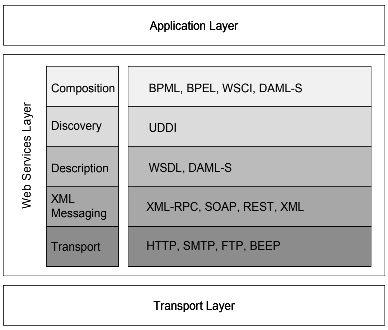


Figure 2 Web Service Protocol Stack

Figure 2 describes the Web service protocol stack which is commonly used within SOAs. For Web services to be successful, they must be able to truly provide interoperability in a manner that is conducive to running a business or producing products that can effectively leverage Web services technology. Thereby they utilize the knowledge of service specification [3].

Composability means that SOAs provide the developer a new way of application deployment by using services newly discovered. Besides, a composition can be carried out dynamically, i.e. at runtime. Composing many services to create a common interface encapsulating the complexity associated with the atomic services is called orchestration.

Orchestration of web services is a concept for sequencing and synchronizing execution of web services. The software part that implements the application logic necessary to orchestrate atomic services is called orchestration engine. An orchestration engine can be concerned as a workflow engine and is responsible for the sequential execution of atomic processes. Orchestration engines do not consider the conversation patterns required to invoke a web service in order to execute a composed process.

Choreography is a complementary technique to service orchestration. Choreography focuses the rules and defines the messages and interaction sequences to occur in order to execute an atomic service through particular service interfaces [7]. WS-BPEL, for instance, is an orchestration language, not a choreography language. Essentially, the primary difference between orchestration and choreography is scope. A choreography model provides a larger scope, encompassing all parties and their associated interactions (e.g. a peer to peer model). An orchestration model is between two participants (specifically focusing on the view of one participant) [14].

There are many means for implementing SOAs such as Web services[3] or CORBA[4]. Web service technology is indeed the preferred implementation vehicle for SOA and the implementation approach described in this article will make use of them. Besides, standardized Web services are also satisfying QoS[5] requirements such as performance and availability [5] [6].

Summarizing it can be said, that the added value of a service-oriented architecture does not lie in the underlying enterprise technologies. Actually it is in the ability of the application to respond to change, and optimize services utilizing different technologies as vehicle for achieving maximum flexibility and agility of the according software system.

IV. Approach

To avoid a completely technical driven approach which basically leads to a loss of scope due to the strong linkage between logical and technical contents the authors decide for a business model driven approach. Utilizing the best-of-breed method this approach intends to maintain the established software systems of each company and integrate them into the SOA.

The basic approach is described in Figure 3. It illustrates the SOA in context to the above mentioned project objectives. Generally the authors structured the conceptual design of the SOA into three stages:

- initial modeling of business processes
- deployment into overall architecture
- analytic feedback to process owner

Within these stages the five subtasks of developing a data integration platform, integrating the existing business process, integrating the change management platform, integrating the flexibility evaluation platform and of providing an easy-to-use interface have to be included. Since the focus of this paper lays on the integration of the flexibility and change management platform, these will be described in the following.

[3] SOA is often referred in literature as Web services. Effectively, Web services are only means for implementing SOA.
[4] Common Object Request Broker Architecture
[5] Quality of Service

In the beginning, existing business processes have been collected and stored automatically – if available - and manually in the persistence data base (cp. Figure 3) from the participating companies. It is an important aspect regarding the acceptance of integrated solutions to keep the already existing business process data untouched.

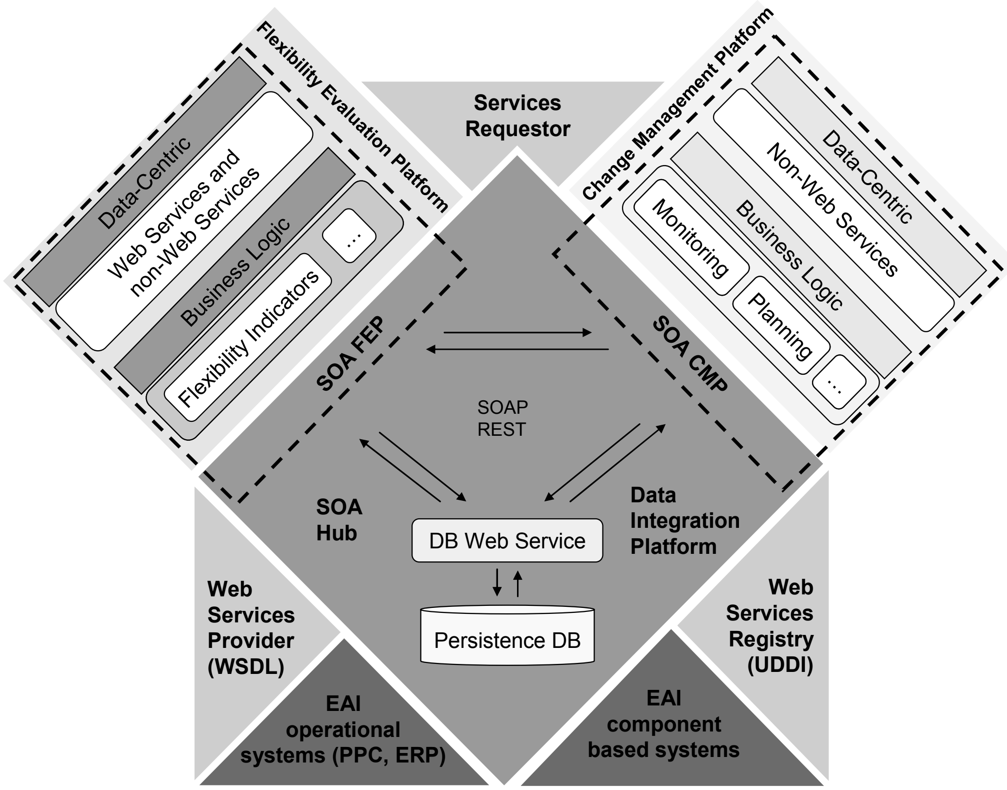


Figure 3 SOA for integrating flexible manufacturing and change management processes

Both in the flexibility evaluation platform and the change management platform these data will be used for creating the internal business logic. Since the objectives here have to cover a "near real-time" flexibility measurement the relevant data has to be permanently available. To calculate the flexibility of the production system raw data deriving from the operational systems (PPC, ERP) are seamlessly "live" integrated into the persistence data base.

The flexibility evaluation platform and the change management platform have been encapsulated through Web services into service-oriented architectures. The communication between these platforms is enabled through standard Web service protocols like SOAP (if activity based) and REST (if resource based).

An exemplary scenario can be the following. A customer has bought a single drilling machine at a vendor who is using the X-Change platform and realizes due to changed market demands that the requirements for this machine have changed. Consequently he wants to change the setup of the drilling machine accordingly to the new demands. Initially the Web server (services requester) searches for the suitable services in the Web service registry. In case the registry does not know the requested service, it cannot complete the request, in case it does, it will return the service location. The request is then forwarded to the appropriate application. The application logic submits a data base query. The query is processed and will be returned afterwards to the application. This process is asynchronous in order to avoid idle time in the application i.e. applications should be capable to process other queries for example monitoring of the "live" flexibility of a system. The requested information is then processed by the application logic to fulfill the user request. This process itself could be a composed process which includes collaboration between the flexibility and change management platform. After the result is computed, the application passes the result back to the Web server. At last the result is forwarded to the customer. Figure 4 illustrates the abstract information flow between the participating components.

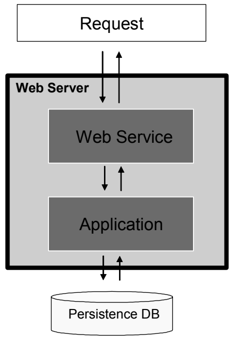


Figure 4 Abstract information flow

By encapsulating the platforms as Web services, businesses can fundamentally transform their ability to interact and engage with both internal systems as well as applications from potential customers and partners utilizing SOA technologies.

V. Conclusion

With increasing individualization of customer product requirements over time, the growing complexity of production management and the ever faster changing environment causes the number of products and variants to steadily rise. At the same time, the possibility to forecast the production determining constraints decreases. Flexibility of production systems is nowadays considered as an important performance factor to predict production constraints. It offers companies not only an important precondition to maintain their competitiveness but also provides new competitive advantage.

Utilizing the knowledge about the relation between business processes and product systems, novel technologies like SOA can close the gap between business-oriented and technical-oriented contents. Consequently, SOA is about getting effective reuse from IT infrastructure and applications rather than from manufacturing infrastructure and machinery, but the benefits are the same: business flexibility, lower cost of change, simplified management.

The approach discussed in this paper focused on the integration of the flexibility and change management platform on IT level but left out the provision of an easy-to-use human-machine interface. This work has to be completed in the forthcoming time.

Acknowledgment

This work has been partly funded by the European Commission through IST-NMP-Project X-Change, Flexible Change Management for the Factory of the Future, (IST & NMP research area: 3.4.5 IST-NMP-1 Integrating Technologies for the Fast and Flexible Manufacturing Enterprise, http://www.x-change-project.net/). The authors wish to acknowledge the Commission for their support. They also wish to acknowledge their gratitude and appreciation to all the X-Change project partners, in particular to LMS (Laboratory for Manufacturing Systems and Automation) from the University of Patras in Greece, to FZI (Research Center for Information Technologies) of Karlsruhe in Germany and to the enterprises Vincent Industrie and Frigoglass for their contribution during the development of various ideas and concepts presented in this paper.

References

[1] Wiendahl, H.-P. and Hernández, R. and Grienitz, V.: *Planung wandlungsfähiger Fabriken*, ZWF 97 (2002) H. 1-2.
[2] Kaluza, B. and Blecker, Th.: Flexibilität – *State of the Art und Entwicklungstendenzen. Erfolgsfaktor Flexibilität. Strategien und Konzepte für wandlungsfähige Unternehmen*, Berlin 2005.
[3] Tsai, W.T. and Chun Fan and Yinong Chen and Paul, R. and Jen-Yao Chung: "Architecture Classification for SOA-Based Applications", *Ninth IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC 2006)*, 24-26 April 2006, pp. 8.
[4] Erl, T., *Service-Oriented Architectures: Concepts, Technology, and Design*, ISBN 0-13-185858-0, New York, Prentice Hall, 2005.
[5] Menascé, D.A.: "QoS Issues in Web Services", *IEEE Internet Computing*, vol. 6, no. 6, 2002, pp. 72 – 75.
[6] Menascé, D.A.: "Composing Web Services: A QoS View", *IEEE Internet Computing*, vol. 8, no. 6, 2004, pp. 88 – 90.
[7] Jammes, F. and Smit, H. and Lastra, J. L. M. and Ivan M. Delamer, "Orchestration of Service-oriented Manufacturing Processes", *Proc. of the 10th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2005)*, September 19-22, 2005
[8] Zhou, B. and He, Z.J. and Tang, J.F.: "An adaptive model of virtual enterprise based on dynamic web service composition", *The Fifth International Conference on Computer and Information Technology (CIT 2005)*, 2005, pp. 284 – 289.
[9] Ivanov, K. and Stähler, D.: *Prozessmanagement als Voraussetzung für SOA*, OBJEKTspektrum Heft, March, 2005.
[10] Myerson, J.: *Close enterprise system gaps with multiple SOAs*, March, 2005.
[11] Myerson, J.: *Consolidate your SOAs as a three-dimensional integration hub to improve speed and reliabilitys*, March, 2005.

[12] Bengtsson, A. and Westerdahl, L.: “Secure Choreography of Cooperating Web Services”, *IEEE International Conference on Web Services (ICWS 2005)*, 2005, pp. 152-159.
[13] Huy, H.P. and Kawamura, T. and Hasegawa, T.: "Web Service Gateway - A Step Forward to E-Business," *IEEE International Conference on Web Services* (ICWS 2004), 2004, p. 648.
[14] OASIS Web Services Business Process Execution Language (WSBPEL), 2006:
[15] Krappe, H. and Rogalski, S. and Sander, M.: “Challenges for Handling Flexibility in the Change Management Process of Manufacturing Systems”, *The second annual IEEE Conference on Automation Science and Engineering CASE 2006 (accepted paper)*
[16] Puliert, E. and Taylor, H.: *Understanding Enterprise SOA*, ISBN 1-932394-59-1, Connecticut; Manning, 2006.
[17] Mayo, S. and Hedim, M.; *Why Are Services-Oriented Architectures Relevant to IT and Business Executives Alike?*, 2004.
[18] Chryssolouris, G.: Manufacturing Systems-Theory and Practice, 2nd Edition (Springer-Verlag), 2005
[19] Chryssolouris, G., "Flexibility and Its Measurement", *CIRP Annals*, (Vol.45, No.2, 1996), pp. 581-587

Survey on News Mining Tasks

Hassan Sayyadi
Web Intelligence Laboratory
Sharif University of Technology
sayyadi@ce.sharif.edu

Sara Salehi
Web Intelligence Laboratory
Sharif University of Technology
sarasalehi@ace.tju.ir

Hassan AbolHassani
Web Intelligence Laboratory
Sharif University of Technology
abolhassani@sharif.edu

Abstract—**Nowadays, there are plenty of online websites related to news. Hence, new technologies, tools and special search engines are created for having access to the news on these websites. Online news is a special type of public information which has exclusive characteristics. These characteristics contribute news engines tasks such as discovering, collecting and searching to be different with similar tasks in traditional web search engines. Clustering plays conspicuous role in news engines tasks. In this paper we study various tasks in news engines and also focusing on clustering applications in them.**

***Index Terms*— News, Clustering, News Retrieval, News Mining**

I. Introduction

NOWADAYS, there are plenty of online websites related to news. Traditional news agencies give their information to their clients via corresponding websites. Hence, new technologies, tools and special search engines are created for having access to the news on these websites. As an instance, News Feeder softwares, RSS standard and Google news website (using 4500 source news) can be mentioned.

Furthermore, online news is a special type of public information which has exclusive characteristics. These characteristics contribute news engines tasks such as discovering, collecting and searching to be different with similar tasks in traditional web search engines. The existence of numerous reliable news sources (high trust) and fast news update are the two most important differences.

News engines provide many services and contain various tasks, the quality of each task can affect the other tasks quality. The most important tasks are:

- Collecting News
- News Retrieval
- Categorizing Search Result
- Summarization
- Automatic Event Detection

Moreover, Clustering is a practical and useful solution for all of news mining tasks and lead to efficient and better results.

In this paper we study various tasks in news engines and also focusing on clustering applications in them. The remainder of this paper is organized as follows: In sections 1 and 2 we will explain collecting news and news retrieval. In the next section we focus on categorizing search results. Next, in section 4 and 5 summarization and automatic event detection will be explained.

II. Collecting News

The first necessity of each news service is to collect news to perform other tasks. Like other web search engines, news engines categorize to three groups, each uses one strategy to collect news corpuses:

1) The engines in which news are submitted to the system by humans manually.
2) Meta-search engines
3) The engines which crawl and discover news sources in the internet and extract news articles automatically.

The engines such as Vivisimo[1] and NewsInEssence[2] are the meta-search engines which don't have collecting process. In these engines, after receiving a user query, query will pass to the other search engines and their output will treated and showed to the user. On the other words, these engines receive the ranked news related to the user's query from other engines via libraries, web services or by processing other engines output pages.

The third group uses different methods for collecting news from available resources in internet. For this type of engines, one of the first and simplest practical ways is to generate news pages URL automatically. For example, a news website contains some fixed groups. Each group includes some news web pages which have a URL with a fixed format. As an instance, news in sport group has an address in the form of http://example.org/sport/n123.html. Consequently, by knowing different groups in each news website, it is possible to create all addresses just by changing news number from 1 to the number of the last news web page. This can help us in collecting the news. Because the news has distinct parts as date, title, and body which are remarkable in other tasks such

Manuscript received October 13, 2006. This work was supported in part by the Web Intelligence Research Laboratory, Sharif University of Technology.

1 http://www.vivisimo.com/
2 http://www.newsinessence.com/ili

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 219–224.

as retrieval, one of the main weaknesses of this method is its disability for extracting these parts. Hence, the format of collected news pages of each source should be detected for extracting each part. Therefore, for collecting news with this method, the human's helps and manual operations are needed. By virtue of this weakness, the way of automatic news extraction for the whole process including news corpuses and their identifications such as date, title and body is much more concerned and different methods are proposed in this way.

Authors of [12] proposed novel automatic news extraction from news sites using Tree Edit Distance measure. Since the structure of a web page can be nicely described by a tree (e.g., a DOM tree), they have resorted to the concept of tree edit distance to evaluate the structural similarities between pages. Intuitively, the edit distance between two trees TA and TB is the cost associated with the minimal set of operations needed to transform TA into TB. To extract the desired news, their approach recognizes and explores common characteristics that are usually present in news portals. Their approach relies on the basic assumption that the news site content can be divided in groups that share common format and layout characteristics. This set of common layout and format features is called a template. According to this approach, the extraction task is performed in four distinct steps: (1) page clustering, (2) extraction pattern generation, (3) data matching and (4) data labeling (see figure 1).

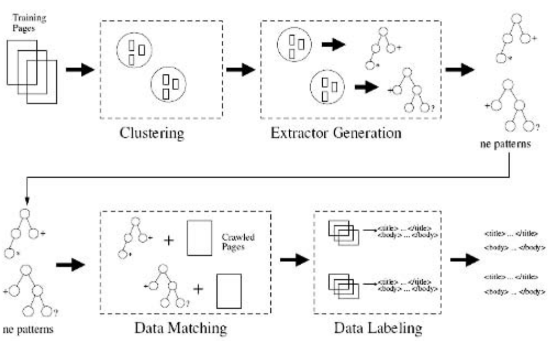


Fig. 1. Overall exteracting steps

The first step takes as input a previously crawled set of pages (a training set) and generates clusters of pages that share common formatting/layout features, i.e., share the same template. The similarity of templates which used for clustering is the Tree Edit Distance measure. Each cluster is later generalized into an extraction structure for a template, in the extraction pattern generation step. After extracting common patterns in templates, the next step is data matching which uses extracted patterns for classification of the newly added news pages to find their templates for extracting news information from those pages. Then, in the data labeling step, for each pattern they will find various parts of each template. So they try to find body, title and date for each pattern. In other words, the passage elected to be the body of the news is the longest one with more than 100 words. Further, the passage selected to be the title is one that has ranges from 1 to 20 words, has a maximum intersection with a body passage, and is the closest one to the body. The intuition behind the title selection is that most of the times the title is placed near the body and its terms usually appear in the news body.

By the advent of RSS standard and related technologies, automatic news extraction methods are no longer useful.

III. RETRIEVAL

After collecting news corpuses, the next step is retrieval task. In the field of news retrieval, most of the engines use traditional ways of information retrieval such as TF-IDF and PageRank. However, the special characteristics of news such as time, topic and importance have strong influence in news ranking of retrieval step. According to the special properties of news, some criteria are proposed for ranking which are appropriate in this domain. One of the important criteria is the time of news. The more the news is new, the more it is significant and the hot news is more attractive to news readers. On the other hand, the news rank can be affected by its cluster because the importance of a news is arises from the number of news related to it. Hence, the more the number of news about one subject is, the more the news is hot. On the other words, the more one cluster is large, the more its news are hot. For this reason, most of the time, news is clustered and the size of each cluster shows the importance of its news.

The affect of clustering in information retrieval was first studied by van Rijsbergen clustering theory [21]. This theory says that documents which are similar to each other will have similar results for similar queries. On the other hand, related documents are much more similar to each other than unrelated documents. Relevant to this theory, clustering can be used before retrieval which is preprocessed like [16] which creates a list for documents set. So by retrieving pages from each cluster it seem rational to retrieve other pages from respect cluster and list in result related to query.

In commercial area, there are many works done for ranking and retrieving news, but there are a few in researches. The few collegiate researches in this field are done in [1,3] and in [11] for finding news articles on the web that are relevant to news currently being broadcast. Gulli et. al [1] proposed the model for ranking news and source news. They assume 5 specifications for their model:

- Ranking for News posting and News sources: the algorithms should assign a separate rank for news articles and news sources.
- Important News articles are clustered: more important news is announced by the large number of news sources and the more the size of cluster is big, the more its news is significant.
- Mutual reinforcement between news articles and news source: hot news is announced by important source and important source announce hot news.
- Time awareness: The importance of a piece of news changes over the time. They are dealing with a stream of information where a fresh news story should be considered more important than an old one.

- Online processing: The time and space complexity of the ranking algorithm allows online processing, i.e. at some time the complexity can depend on the mean amount of news articles arriving but not on the time since the observation started.

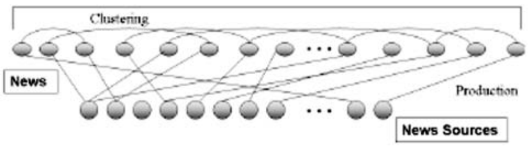


Fig. 2. Model specifications

Based on these criteria, and which is showed in figure 2, they proposed a repeated model derived from the mutual reinforcement between news and news sources. In this model the rank of a news source and news article are computed as follow:

$$R(s_k,t) = \sum_{s(n_i)=s_k} e^{-\alpha(t-t_i)} R(n_i,t) + \sum_{s(n_i)=s_k} e^{-\alpha(t-t_i)} \sum_{t_j>t_i \text{ and } S(n_i)\neq s_k} \sigma_{ij} R(n_j,t_i)^{\beta}$$

$$R(n_i,t_i) = [\lim_{\tau\to 0^+} R(S(n_i),t_i-\tau)]^{\beta} + \sum_{t_j<t_i} e^{-\alpha(t_i-t_j)} \sigma_{ij} R(n_j,t_j)^{\beta}$$

Where *R(s,t)* is the rank of news source *s* , and similarly, *R(n,t)* is the rank of news article *n* which has been released at time *t*. The value α is obtained from the half-life decay time $e^{-\alpha\rho}=1/2$, that is the time required by the rank to halve its value, with the relation *ρ* which is a model parameter. *ß* is also a model parameter. *S(n)* indicates the news source for news article n and *бij* shows the correlation between two news article *i* and *j* which is obtained from the similarity of snippets.

IV. Categorizing Search Result

One of the prominent problems of current search engines is showing output result in the form of ranked list without categorizing search results. If the user's query is exact enough, the results are exact and few. But for general and vague queries, the results range is very wide. On the other hand, the queries generally contain three words and users usually check the first three pages of the output result [5]. For example, assume that you want Google search engine to find related results for "Jaguar" query. "Jaguar" is a name of kind of cat and is also the drink brand, if you want the results related to the first meaning you should see all the results and extract your desire results, but due to the fact that users commonly check the first three pages, they never reach to their desired results. Consequently, it is necessary to put the output results of every query in different groups and give the categorized results to the user. By doing this method, user will have a comprehensive vision about the results. On the other hand, due to the fact that the first pages always contain the results related to one group, algorithms based on link analysis deprive users from discovering about different groups of results. Consequently, categorization results in better display and easier exploration of search results.

Classification and clustering are the main methods of search result categorization. Clustering on the results of search engines has obvious differences with traditional text clustering. One of these differences is the existence of links between web pages. Also, Due to the fact that search result clustering is an online process, fast computation is needed. The final difference is that clustering is based on small snippets instead of whole document. According to [18] good characteristics for clustering on the search results are defined as bellows:

- No necessity for all pages to be clustered
- Cluster Overlapping
- Incremental clustering

One of the reasons that commercial search engines don't do clustering is the high execution time cost. Some search engines such as Altavista[3] do aggregation in a very simple way. In "Northern Light"[4], results are classified to some custom folders. This classification is out of any intelligence and it is done on the basis of specifications such as page type, language, domain, and site and… Better commercial examples are included "Kartoo"[5], "Grokker"[6], "Mooter"[7], and specially "Vivisimo" which uses clustering algorithms. These engines have suitable qualification in respect to user interface. "Clusty"[8] is also use Vivisimo, but its clustering method is not obvious enough.

In spite of many usages of clustering in dynamic engines [8,9,14,15,16,17,19], Classification may be used in higher levels or just in one level. The reason for that idea is the need of classification to manual operations, and different groups related to the different queries. Owing to these constraints, in this section we are focusing on clustering.

Another important task which should be done after clustering is cluster labeling. Labels are essential for each cluster, so the better the labels describe clusters, the better the quality of clustering algorithm is. Referring to [8] there are two specifications for selecting cluster labels:

1. Label Readability
2. Label describing respective cluster accurately

Regarding the aforementioned statements, clustering approaches fall into two categories:

- Document-based techniques

[3] Http://www.altavista.com/
[4] Http://www.northernlight.com/
[5] Http://www.kartoo.com/
[6] Http://www.grokker.com/
[7] Http://www.mooter.com/moot/
[8] Http://www.clusty.com/

- Label-based techniques

Document-based techniques are such traditional methods that do clustering by defining similarity measures between document specifications like keyword vector. After that some words and sentences of the documents in each cluster are extracted and shown to the users as a cluster label like [15,16,19]. Clusters created by this technique do not have overlap and the quality of clusters label is under the influence of clustering accuracy. On the other hand, controlling the number of clusters and defining the similarity thresholds which determine the quality of cluster is difficult. Consequently, labels quality is not satisfactory for users. For this reason, these methods are no longer used for clustering search engines results. Works done in Document-based techniques are different from each other in two aspects:

1) Clustering algorithm.
2) Clusters and documents distance measure

Since the output shown to the user is hierarchical, a tendency for using hierarchic algorithms is high. This causes less runtime overhead for creating hierarchy (because the output is in the form of tree itself) but its quality is less than other algorithms.

In Label-based techniques informative words and expressions like high frequent words are extracted from news corpuses via statistical analyses, and among these candidate labels, those which result in better clusters are selected as final clusters labels. In this approach each label creates one cluster and each page which contains that label fall into respective cluster. This approach results in overlapped clusters. Due to the fact that one news can point to several events, overlapping seems more rational.

One of the Document-based methods is the clustering used in [9]. The first step was to remove the stopwords from each article. A stopword occurs so often that it creates no significance to a particular document For instance, the removal of the article “a” and the word “however” does not hinder uniqueness regardless of how often they appear. The second step in data cleansing was converting the resulting documents, containing nonstop words, into stemmed words. After that, they used the Porter stemming algorithm which is a process for removing the commoner morphological and in flexional endings from words in English to prevent the program from treating the word "go" and "went" differently.

The final step involved the extracting of the n most frequently occurring words in each article. If *S1* and *S2* are the sets which includes these extracted words of two news, the similarity between two news is defined as bellow:

$$Sim(n_1, n_2) = \frac{|s_1 \cap s_2|}{n}$$

$$Dissim(n_1, n_2) = 1 - Sim(n_1, n_2)$$

Where *Dissim(n1,n2)* is the distance criterion of two news. By virtue of these criteria, K-Nearest Neighbor is done on news. Single-link algorithm is also used in that paper, but the combination of both algorithms leads to the better results.

On the other hand, Authors of [8] proposed efficient Label-based model by using novel criteria for phrase ranking and Named Entities. This method indicates that the TF-IDF criterion is influenced by term frequency and doesn’t sufficiently reject high frequency terms. Hence, some new criteria are proposed which are much more efficient. We consider the significance of the labels and propose new Local and Global Factors. In traditional method of TF-IDF, TF defined as local factor while IDF defined as global factor. In this paper, two new and efficient local factors named LRDF and OLF are computed as follow:

$$LF_i^{LRDF} = \log(1 + DF_{R,i})$$

$$LF_i^{OLF} = DF_{R,i} * \log(\frac{|R|}{DF_{R,i}})$$

$$DF_i^{OGF} = \frac{DF_{r,i} / |R|}{DF_{D,i} / |D|}$$

Where OGF is a global factor. Their experiments showed the combination of OGF with any local factor lead to better results than all combination of local factors with IDF and the OLF-OGF combination gave the best results. Using Named Entities is very efficient in the filed of news, because an event but a regular web page points to the distinct person, location, organization or date. Consequently, extracting persons or organizations and dates are very productive. The difficulty of their work is that they only select named entities as cluster labels but they are not sufficient and named entities can not describe the clusters as well as none phrase.

Vivisimo and Mooter, two of the best engines, are also use Label-based techniques for their clustering algorithm. So we can conclude that label-based approaches are much more applicable in search results clustering. Furthermore, user interface for result presentation in engines which performing result categorization (Vivisimo, Kartoo, Grokker) is important. A good study about this concept is done in [20].

V. SUMMARIZATION

As we mentioned, one of the usages of clustering is in summarizing news. One of the successful works done in this field is proposed in [7]. In this research an engine named “SimFinder” is implemented which can give summarization of some news document to the users by clustering paragraphs. The reason for choosing paragraph in clustering is that more specialized information can be utilized in working with smaller units of text (sentences or paragraphs). They first identified 43 features for text which could efficiently extract from the text and that could plausibly help determine the semantic similarity of two short text units. Finally they select 11 of the 43 features by using data mining tasks. Afterwards, paragraphs are clustered by using these features. They cast the clustering problem as an optimization task and seek to minimize an objective function ϕ measuring the within-cluster dissimilarity in a partition:

$$\phi(\rho)=\sum_{i=1}^{k}[\frac{1}{|C_i|}\sum_{x,y\in C_i \text{ and } x\neq y} d(x,y)]$$

Then MULTIGEN goes beyond sentence extraction into reformulation and analyzes the sentences in each cluster produced by SIMFINDER and regenerates instead a new sentence containing just the information common to almost all sentences in a cluster.

Moreover, another engine named NewsInEssence [4], a fully deployed digital news system. A user selects a current news story of interest which is used as a seed article by NewsInEssence to find in real time other related stories from a large number of news sources. The output is a single document summary presenting the most salient information gleaned from the different sources. NewsInEssence first perform focused crawling which start from given news page. Then, some keywords which are more descriptive will be extracted from crawled pages contents. Afterward, keywords will be passed to some search engines and their result will be fetched. Finally, the summarization of fetched pages will be displayed to the users.

VI. Automatic Event Detection

Another task in news engines is automatic event detection [6,2,13]. Authors of [6] generate sentence level clusters using hierarchical algorithms such as single-link, complete-link, and GroupWise-average. By keeping in mind that news can point to different events, they proposed that by clustering news in sentence level, each cluster point to an event. In this paper, WORDNET is used to heighten the efficient of comparison between words and sentences. It also use one learning automata for concerning the location of sentences in the document in clustering. Inasmuch as sentences relating to one event should be near each other, clustering and summarization are done over the sentences related to one event.

In [10], there is also one method for dividing news to its event parts and assigning one topic to each. The focus of this model is on automatic speech recognizer (ASR) scripts.

Segmentation system is a two stage process: the first stage hypothesizes boundaries, and the second stage removes boundaries. The first stage of the segmentation system uses a binary decision tree based probabilistic model to compute the probability of a boundary at every point in the ASR transcript that has been labeled a non-speech event. The features proposed for the decision tree are extracted from finite windows to the left and right of the current point. The features used by the tree are selected automatically. After the story boundaries have been hypothesized, a second stage (within the deferral period) removes some of them in order to reduce the false-alarm rate. The second stage uses the document-document similarity score of our detection system to determine if adjacent stories are similar topically, and reject the hypothesized boundary between them. The refinement step is applied iteratively.

Furthermore, a topic detection algorithm for detected stories proposed which is an incremental clustering algorithm that employs a novel dynamic cluster-dependent similarity measure between documents and clusters used for topic detection algorithm and decision tree segmentation which is a classification model. As soon as the document is added, its similarity degree with other clusters is measured. If the similarity degree of that document to one cluster is higher than defined threshold, then it entered that cluster.

The similarity measure used to obtain similarity of document d^1 and d^2 is computed as follow:

$$Sim(d^1,d^2)=\sum_{w\in d^1\cap d^2} t_w t_w^2 idf(w,cl)$$

Where t_w^i is the term count of word w in document i and *idf(w,cl)* is the cluster-dependent inverse document frequency of word w in cluster cl.

VII. Conclusion

In this work we studied various tasks and services in news engines and survey on models and approaches applied in each task. Furthermore, we focused on clustering usages in each task in detail. As showed in this paper, clustering has many productive applications in all steps. For example, clustering helps to automatic news collection and cluster size is an important measure for news ranking. Clustering leads to a better display and easier explore of the search results. Consequently, clustering contributes to many efficient methods and results in news summarization and event detection.

References

[1] G. M. D. Corso, A. Gulli, F. Romani, “Ranking a stream of news”, In: WWW ’05: Proceedings of the 14th international conference on World Wide Web, New York, NY, USA, ACM Press (2005) 97–106

[2] M. Atallah, R. Gwadera, W. Szpankowski, ”*Detection of significant sets of episodes in event sequences*”. icdm 00 (2004) 3–10

[3] N. Maria, M. J. Silva, ”*Theme-based retrieval of Web news*”, Lecture Notes in Computer Science 1997

[4] D. R. Radev, S. Blair-Goldensohn, Z. Zhang, R. S. Raghavan, ”*Interactive, domain-independent identification and summarization of topically related news articles*”.

[5] C. Silverstein, M. Henzinger, H. Marais, and M. Moricz, ”*Analysis of A Very Large Web Search Engine Query Log*“, SIGIR Forum, 33(1), 1999.

[6] M. Naughton, N. Kushmerick, J. Carthy, ”*Clustering sentences for discovering events in news articles*”, In: ECIR. (2006) 535–538

[7] V. Hatzivassiloglou, J. Klavans, M. Holcombe, R. Barzilay, M. Kan, K. McKeown, “*Simfinder: A flexible clustering tool for summarization”*, (2001)

[8] H. Toda, R. Kataoka, “*A search result clustering method using informatively named entities”,* In: WIDM ’05: Proceedings of the 7th annual ACM international workshop on Web information and data management, New York, NY, USA, ACM Press (2005) 81–86

[9] N. A. Shah, E. M. ElBahesh, “*Topic-based clustering of news articles*”, In: ACM-SE 42: Proceedings of the 42nd annual Southeast regional conference, New York, NY, USA, ACM Press (2004) 412–413

[10] S. Dharanipragada, M. Franz, J. McCarley, S. Roukos, T. Ward, ”*Story segmentation and topic detection in the broadcast news domain*”, (1999)

[11] M. Henzinger, B. Chang, B. Milch, S. Brin, "*Queryfree news search*", (2003)

[12] D. Reis, P. Golgher, A. Silva, A. Laender, "*Automatic web news extraction using tree edit distance*", (2004)

[13] Li, Z., Wang, B., Li, M., Ma, W.Y., "*A probabilistic model for retrospective news event detection*", In: SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM Press (2005) 106–113

[14] Z. Jiang, A. Joshi, R. Krishnapuram, and L. Yi., "*Retriever: Improving Web Search Engine Results Using Clustering*", In Managing Business with Electronic Commerce 02.

[15] O. Zamir and O. Etzioni, "*Grouper: A Dynamic Clustering Interface to Web Search Results*". In Proceedings of the Eighth International World Wide Web Conference, Toronto, Canada, May 1999.

[16] M. A. Hearst and J. O. Pedersen, "*Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results*", in Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR96), 1996, pp 76-84.

[17] H. Chen and S. Dumais, "*Bringing Order to The Web: Automatically Categorizing Search Results*", in Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems, pp. 142–152, 2000.

[18] Y. Wang and M. Kitsuregawa, "*Link Based Clustering of Web Search Results*", in Second International Conference on Advances in Web-Age Information Management (WAIM), 2000.

[19] H. Zeng, Q. He, Z. Chen, W. Ma, and J. Ma, "*Learning to Cluster Web Search Results*". In Proceedings of ACM SIGIR '04, 2004.[19] Douglas Cutting, Jan O. Pedersen, David Karger, and John W. Tukey. "Scatter /Gather: A Cluster-Based Approach to Browsing Large Document Collections". In Proceedings of SIGIR'92, pages 318-329, Copenhagen, Denmark, June 21-24 1992.

[20] W. Rivadeneira, and B. B. Bederson**,** "*A Study of Search Result Clustering Interfaces: Comparing. Textual and Zoomable User Interfaces*". University of Maryland HCIL Technical Report, HCIL-2003-36

[21] V. Rijsbergen, C. J., "*Information Retrieval*". London: Butterworths; 1979.

Configuring and Designing Replication in Active Directory

Hemant Kumar Arora

Abstract – **Active Directory is widely used across the various organizations for Windows® infrastructure authentication. Organizations may vary from very small having only one office to very large spread across the globe with various offices. Domain controllers are the heart of Active Directory and store its database. For Active Directory to function properly, its database should be consistent across all the domain controllers which is achieved using replication, hence Replication is integral part of Active Directory. For very small organizations or organizations that are well connected using high bandwidth network links, replication doesn't create any major problem and happens smoothly without much of the configuration needed. But for the organizations which are very large and have a large number of small offices spread across the globe with limited connectivity, configuring replication is a very important and tough task. If the replication is not properly configured in such organizations, whole deployment of Active Directory may fail. Configuring replication involves mapping the Active Directory to the physical network of the organization which is achieved by configuring sites, subnets, site links and site link bridgeheads. This paper discusses the concept and need of replication in Active Directory and configuration and design of its replication for large organizations. This paper also discusses the best practices to configure replication among large organizations and illustrate it by demonstrating it for a dummy organization.**

I INTRODUCTION

Generally Replication may be defined as a duplicate copy of similar data on the same or a different platform or system. AD(Active Directory) is a database which contains various objects classes like users, groups and computers and these object classes have attributes like first name, last name and memberof for user class. Domain controllers are Windows® servers which stores AD database and provide authentication services for users. If any object class attribute value is changed on any of the domain controller it is replicated among all the domain controllers in the AD to keep the database consistent. AD uses multimaster replication scheme which means that any change to an AD object can happen to any domain controller. Replication is the process which transfers the changes to AD database between domain controllers. The domain controllers may be located on the same site or may be located across the globe connected through a low bandwidth network. Although the goals of replication are quite simple, the real world constraints of the network connections between domain controllers cause many limitation that must be accommodated. AD uses the concept of sites to map to an organizational physical network. Site can be defined as a collection of well connected computers. AD site is merely based on the organizational physical network and there is no specified relation between AD domains and AD sites. An AD site can contain multiple domains. Alternatively, a single AD domain can also span across multiple sites. Replication strategy is primarily based on sites and a well designed replication can help any organization to achieve the following result:

- Minimize the cost of replicating AD data.
- Minimize administrative efforts that are required to maintain the site information.
- Schedule replication that enables locations with slow or dial-up network links to replicate AD data during off-peak hours.
- Optimize the ability of client computers to locate the nearest resources, such as domain controllers and Distributed File System (DFS) servers, reducing network traffic over slow, wide area network (WAN) links and improving logon and logoff processes.

II PURPOSE OF SITE INFORMATION

Site information is not only used by replication infact AD uses site information for many purposes including routing replication, client affinity, system volume replication and DFS. Their brief descriptions are given below:

A. Routing replication

AD uses a multimaster, store-and-forward method of replication. A domain controller communicates directory changes to a second domain controller, which then communicates to a third, and so on, until all domain controllers have received the change. To achieve the best balance between reducing replication latency and reducing traffic, site information controls AD replication by distinguishing between replication that occurs within a site (intrasite) and replication that occurs between sites (intersite).Within sites, replication is optimized for speed data updates trigger replication and the data is sent without the overhead required by data compression. Conversely, replication between sites is compressed to minimize the cost of transmission over WAN links. When replication occurs between sites, a single domain controller per domain at each site collects and stores the directory changes and communicates them at a scheduled time to a domain controller in another site.

B. Client affinity

Domain controllers use site information to inform AD clients about domain controllers present within the same or closest site as the client. For example, a client in the New Delhi site that does not know its site affiliation and contacts domain controller from the Bangalore site. Based on the IP address of

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 225–230.

the client, the domain controller in Bangalore determines which site the client is actually from and sends the site information back to the client. The domain controller also informs the client whether the chosen domain controller is the closest one to it. The client caches the site information provided by the domain controller in Bangalore and queries for the site-specific service (SRV) resource record (a DNS resource record used to locate domain controllers for AD) and thereby finds a domain controller within the same site.

By finding a domain controller in the same site, the client avoids communications over WAN links. If no domain controllers are located at the client site, a domain controller that has the lowest cost connections relative to other connected sites advertises itself (registers a site-specific SRV resource record in DNS) in the site that does not have a domain controller. The domain controllers that are published in DNS are those from the closest site as defined by the site information. This process ensures that every site has a preferred domain controller for authentication.

C. SYSVOL replication

The system volume (SYSVOL) is a collection of folders in the file system that exists on each domain controller in a domain. The SYSVOL folders provide a default AD location for files that must be replicated throughout a domain, including Group Policy objects (GPO), startup and shutdown scripts, and logon and logoff scripts. AD uses the File Replication service (FRS) to replicate changes made to the SYSVOL folders from one domain controller to other domain controllers. FRS replicates these changes according to the schedule that we create during our site topology design.

D. DFS(Distributed File System)

DFS uses site information to direct a client to the server that is hosting the requested data within the site. If DFS does not find a copy of the data within the same site as the client, DFS uses the site information in AD to determine which file server that has DFS shared data is closest to the client.

III AD Replication Concepts

Before we discuss the Replication design we should describe the Basic AD replication terms which are given below:

A. Connection object

A connection object is an AD object that represents a replication connection from one domain controller to another. A domain controller is a member of a single site and is represented in the site by a server object in AD. Each server object has a child NTDS Settings object that represents the replicating domain controller in the site. The connection object is a child of the NTDS Settings object on the destination server.

For replication to occur between two domain controllers, the server object of one must have a connection object that represents inbound replication from the other. All replication connections for a domain controller are stored as connection objects under the NTDS Settings object. The connection object identifies the replication source server, contains a replication schedule, and specifies a replication transport. The Knowledge Consistency Checker (KCC) creates connection objects automatically, but they can also be created manually. Whenever we change a connection object created by the KCC, we automatically convert it into a manual connection object. The KCC stops making changes to the manual connection object.

B. KCC

The KCC is a built-in process that runs on all domain controllers and generates replication topology for the AD forest. The KCC creates separate replication topologies depending on whether replication is occurring within a site (intrasite replication) or between sites (intersite replication). The KCC dynamically adjusts the topology to accommodate new domain controllers, domain controllers moved to and from sites, changing costs and schedules, and domain controllers that are temporarily unavailable.

Within a site, the connections between domain controllers are always arranged in a bidirectional ring, with additional shortcut connections to reduce latency in large sites. On the other hand, the intersite replication is a layering of spanning trees, which means one intersite connection exists between any two sites for each directory partition and generally does not contain shortcut connections.

On each domain controller, the KCC creates replication routes by creating one-way inbound connection objects that define connections from other domain controllers. For domain controllers in the same site, the KCC creates connection objects automatically without administrative intervention. When we have more than one site, we configure site links between sites and a single KCC in each site automatically creates connections between sites as well.

Figure 1 shows two sites (New Delhi and Bangalore) with domain controllers that are all in the same domain. The arrows represent possible inbound connections that the KCC creates. Because all AD updates are transferred in a ring within a site and redundant connections exist, all domain controllers can receive updates from all other domain controllers in the New Delhi site, although domain controllers within a site do not necessarily replicate in both directions. For replication to occur between New Delhi and Bangalore, one domain controller in each site has a replication agreement with a domain controller in the other site. Between sites, these replication partners replicate in both directions over a site link that represents the physical WAN connecting the two sites. In Figure 1, domain controllers DC-3 and DC-6 are replication partners between the New Delhi and Bangalore sites.

C. Subnet

A subnet is a segment of a TCP/IP network to which a set of logical IP addresses are assigned. Subnets group computers in a way that identifies their physical proximity on the network. Subnet objects in AD identify the network addresses that are used to map computers to sites.

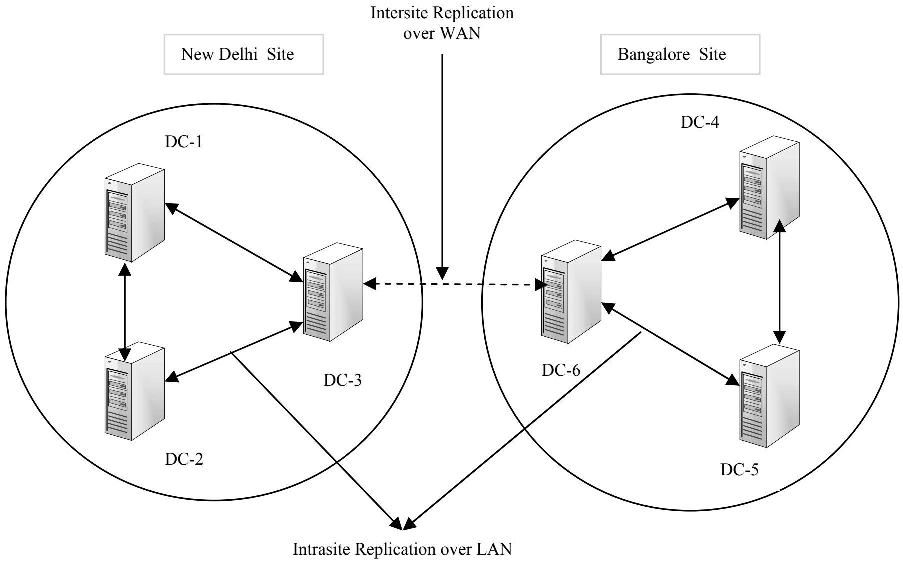


Fig 1: Intersite and Intrasite Replication Connections

D. Site

Sites are one or more TCP/IP subnets with highly reliable and fast network connections. Site information allows administrators to configure AD access and replication to optimize usage of the physical network. Sites are represented in AD as site objects. Site objects are a set of subnets, and each domain controller in a forest is associated with an AD site according to its IP address. Sites can host domain controllers from more than one domain, and a domain can be represented in more than one site.

E. Site link

Site links are logical paths that the KCC uses to establish a connection for AD replication. Site links are stored in AD as site link objects. A site link object represents a set of sites that can communicate at uniform cost through a specified intersite transport.

All sites contained within the site link are considered to be connected by means of the same network type. Sites must be manually linked to other sites by using site links so that domain controllers in one site can replicate directory changes from domain controllers in another site. Because site links do not correspond to the actual path taken by network packets on the physical network during replication, we do not need to create redundant site links to improve replication efficiency.

When two sites are connected by a site link, the replication system automatically creates connections between specific domain controllers in each site called bridgehead servers. The KCC may designate more than one domain controller per site hosting the same directory partition as a candidate bridgehead server. The replication connections created by the KCC are randomly distributed between all candidate bridgehead servers in a site to share the replication workload. By default, the randomized selection process takes place only when new connection objects are added to the site.

F. Site link bridge

A site link bridge is an AD object that represents a set of site links, all of whose sites can communicate by using a common transport. Site link bridges enable domain controllers that are not directly connected by means of a communication link to replicate with each other.

By default, the KCC can form a transitive route through any and all site links that have some sites in common. If this behavior is disabled, each site link represents its own distinct and isolated network. Sets of site links that can be treated as a single route are expressed through a site link bridge. Each bridge represents an isolated communication environment for network traffic.

Site link bridges are a mechanism to logically represent transitive physical connectivity between sites. A site link bridge allows the KCC to use any combination of the included site links to determine the least expensive route to interconnect directory partitions held in those sites. The site link bridge does not provide actual connectivity to the domain controllers. If the site link bridge is removed, replication over the combined site links will continue until the KCC removes the links.

Site link bridges are only necessary if a site contains a domain controller hosting a directory partition that is not also hosted on a domain controller in an adjacent site, but another domain controller hosting that directory partition is located in other sites in the forest. Adjacent sites are defined as any two or more sites included in a single site link.

A site link bridge creates a logical connection between two site links, providing a transitive path between two disconnected sites by using an interim site. For the purposes of the intersite topology generator (ISTG), the bridge implies physical connectivity by using the interim site. The bridge does not imply that a domain controller in the interim site will provide the replication path. However, this would be the case if the interim site contained a domain controller that hosted the directory partition to be replicated, in which case a site link bridge is not required.

The cost of each site link is added, creating a summed cost for the resulting path. The site link bridge would be used if the interim site did not contain a domain controller hosting the directory partition and a lower cost link does not exist. If the interim site contained a domain controller that hosts the directory partition, two disconnected sites would set up replication connections to the interim domain controller and not use the bridge.

G. Site link transitivity

By default all site links are transitive, or "bridged." When site links are bridged and the schedules overlap, the KCC creates replication connections that determine domain controller replication partners between sites, where the sites are not directly connected by site links but are connected transitively through a set of common sites. This means that we can connect any site to any other site through a combination of site links.

In general, for a fully routed network, we do not need to create any site link bridges unless we want to control the flow of replication changes. All site links for a specific transport implicitly belong to a single site link bridge for that transport. The default bridging for site links occurs automatically, and no AD object represents that bridge. The Bridge all site links setting found in the properties of both the IP and Simple Mail Transfer Protocol (SMTP) intersite transport containers implements automatic site link bridging.

IV Designing replication in Active Directory

The first step in designing an effective replication topology is to consult the network group of the organization about the organization physical network infrastructure. For that first we need to create a location map of the organization and list down all the network links, there link speed and available bandwidth and cost for each location. Then we need to collect the information related to the IP addressing scheme of the organization i.e. we need to collect the IP subnet ranges used within each location. Then we need to list the number of users in each location and number of domains used in each location. Once we have collected this information we need to plan the replication topology based on organization network architecture i.e. whether we will use hub topology, hub and spoke topology or complex topology. These Network topologies are shown in figure2. Complex replication topology is most commonly used because in most organization the architecture of network is based on complex topology. In this topology we can designate few hub sites which are connected through high bandwidth link with each other and perform a

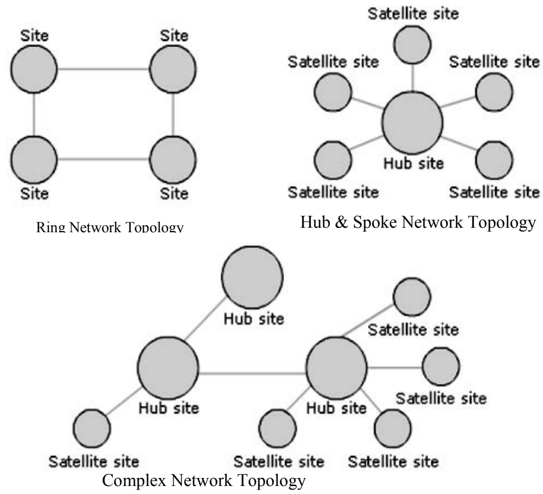


Fig 2: Different Network Topologies

ring and then connect satellite sites to those hub sites. Then we plan domain controllers placement which involves planning forest root domain controller placement, regional domain controllers, operations master role holders, and global catalog servers. The entire designing process is shown in figure 3.

Forest root domain controllers are needed to create trust paths for clients that need to access resources in domains other than their own. Place forest root domain controllers at locations that host datacenters and in hub locations. If users in a given location need to access resources from other domains in the same location, and the network availability between the datacenter and the user location is unreliable, then we can either add a forest root domain controller in the location or create a shortcut trust between the two domains.

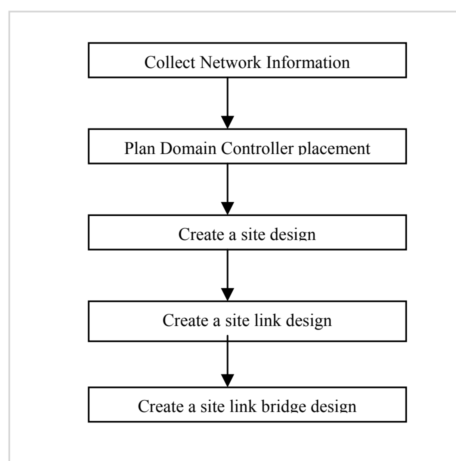


Fig 3: Designing Replication in Active Directory

In a single domain forest, configure all domain controllers as global catalog servers because this does not require any additional disk space usage, CPU usage, or replication traffic. Global catalog servers facilitate user logon requests and forest-wide searches. If our locations include applications that do not deliver optimum response over a WAN link, we must place a global catalog server at the location to reduce query latency. Place global catalog servers at all locations that contain more than 100 users to reduce congestion of network WAN links and to prevent productivity loss in case of WAN link failure.

AD supports multimaster replication of directory data, which means any domain controller can accept directory changes and replicate the changes to all other domain controllers. However, certain changes, such as schema modifications, are impractical to perform in a multimaster fashion. For this reason certain domain controllers, known as operations masters, hold roles responsible for accepting requests for certain specific changes. Place the domain controllers hosting these operations master roles in areas where network reliability is high, and ensure that the PDC emulator and the RID master are consistently available.

Operations master role holders are assigned automatically when the first domain controller in a given domain is created. The two forest-level roles (schema master and domain naming master) are assigned to the first domain controller created in a forest. Additionally, the three domain-level roles (RID master, infrastructure master, and PDC emulator) are assigned to the first domain controller created in a domain.

Place the first domain controller for a domain in a location that has the largest number of users for that domain. Designate a standby operations master for a domain controller that hosts the operations master roles. The standby operations master is a domain controller that we identify as the computer that assumes the operations master role if the original role holder fails. Ensure that the standby operations master is a direct replication partner of the actual operations master.

Once we have designed the domain controller placement we need to decide on which locations to create sites and is described below:

- Create sites for all locations in which we plan to place domain controllers. Refer to the information collected in the Domain Controller Placement worksheet to identify locations that include domain controllers.
- Create sites for those locations that include servers that are running applications that require a site to be created. Certain applications, such as DFS, use site objects to locate the closest servers to clients.
- If a site is not required for a location, add the subnet of the location to a site for which the location has the maximum WAN speed and available bandwidth.

Create a site link design in order to connect our sites with site links. Site links reflect the intersite connectivity and method used to transfer replication traffic. We must connect sites with site links so that domain controllers at each site can replicate AD changes.

To connect sites with site links, we identify the member sites that we want to connect with the site link, create a site link object in the respective Inter-Site Transports container, and then name the site link. After we create the site link, we can proceed to set the site link properties.

When creating site links, we need to ensure that every site is included in a site link. In addition, ensure that all sites are connected to each other through other site links so that the changes can be replicated from domain controllers in any site to all other sites. If we fail to do this, then the KCC generates an error message in the Directory Service log in Event Viewer stating that the site topology is not connected**.** Intersite replication occurs according to the properties of the connection objects.

Setting site link object properties includes the following steps:

- Determining the cost that is associated with that replication path. The KCC uses cost to determine the least expensive route for replication between two sites that replicate the same directory partition.
- Determining the schedule that defines the times during which intersite replication can occur.
- Determining the replication interval that defines how frequently replication should occur during the times when replication is allowed as defined in the schedule.

A site link bridge enables transitivity between site links. Each site link in a bridge needs to have a site in common with another site link in the bridge or else the bridge cannot compute the cost from sites in the link to the sites in the other links of the bridge. Without the presence of a common site between site links, the bridge also cannot establish direct connections between domain controllers in the sites that are connected by the same site link bridge.

By default, all site links are transitive and it is recommended to keep transitivity enabled by not changing the default value of Bridge all site links (enabled by default). However, we will need to disable Bridge all site links to complete our site link bridge design if our IP network is not fully routed or We need to control the replication flow of the changes made in AD.

V XYZ.COM ORGANIZATION –CASE STUDY

Here we will discuss about a dummy organization XYZ.COM and design the AD replication for this organization. The network architecture of the organization is shown in figure 4. From the Network Diagram it is quite clear that this organization uses complex topology which is having three hub sites named Carrolton, AngMoKio and St Genis which are connected through high speed WAN links and each of the hub site is connected through several satellite sites with low bandwidth connections. The first AD server will be installed in St Genis, then dedicated bridgehead servers will be installed in each regional hubs (Carrolton, AngMoKio, St Genis). All other domain Servers will replicate there AD information from these hub locations.

There will be three level of site links which are described below:

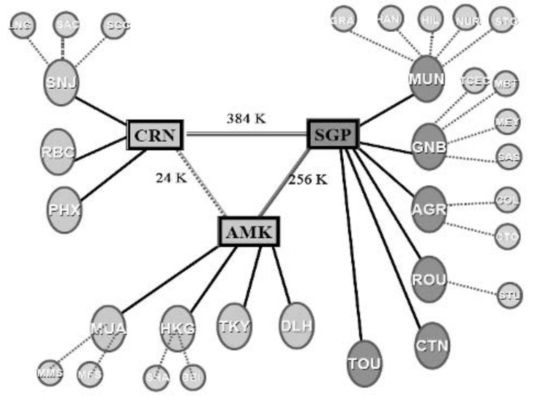


Fig 4: XYZ.COM Organization- Network Architecture

1st Level : International site-links between the 3 hubs locations (SGP, AMK and CRN).
2nd Level : Europe, America and Asia pacific site-links. The 3 sites of the above level acts as the replication Hub.
3rd Level : regional site-links. Represents links between some 2nd level sites and their satellite

The replication topology diagram is shown in figure 5 the replication is designed such as within a site, clients will attempt to communicate with domain controllers in the same site as the client before trying to communicate with domain Controller in any other site.

If a client is on a subnet that is not defined in the directory, it is not considered part of a site, and it selects randomly from all domain controllers for a particular domain.

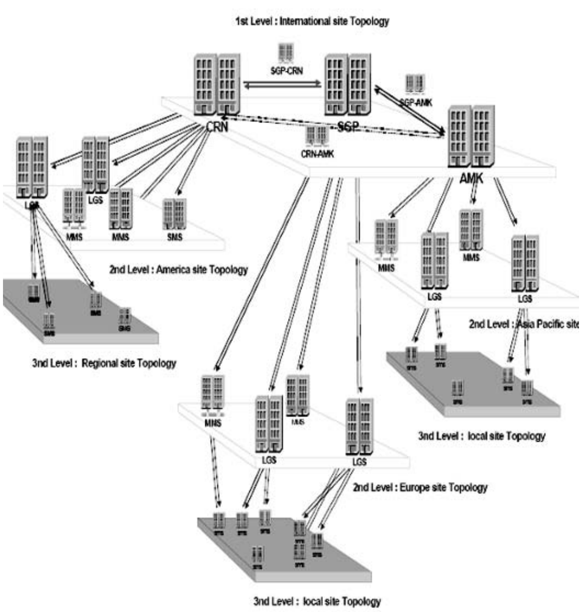


Fig 5: XYZ.COM Organization- Replication topology

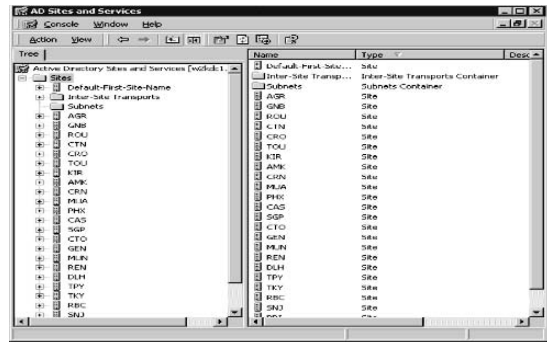


Fig 6: Active Directory Sites and services replication tool

A domain controller must be able to respond to client requests (primarily for network logon and queries to the directory) in a timely manner that suits the requirements of the clients. For best performance in our environment it is recommended to place at least one domain controller in each site that contains more than 10 users or computers.

For XYZ organization very small Branch Offices (1-9 users) are server less sites. In all other sites all DCs will participate in replication with one of the Hub Bridgehead Server. By default the first DC in a site is the Bridgehead server. The Active Directory sites and services tool which will be used to manage these services is shown in figure 6 and using this tool the sites and site links will be created and managed so that KCC can use this information to Replicate AD database.

VI CONCLUSION

In this paper we have discussed about the AD Replication design which is dependent upon the physical network of the organization. We have described the terms like site, site links, site link bridgehead and site link bridges in relation to the AD and defined the whole process and best practices for designing and configuring replication in AD. We discussed the network of dummy organization xyz.com and designed its replication topology. The other application of sites like routing replication, Client affinity, SYSVOL replication and DFS are also described. We also described some other terms in AD like Domain Controller, FSMO roles, Global Catalog Servers etc. After going through this paper anyone can understand the AD replication in a better manner and think of designing the replication even for the complex organization.

References

[1] Microsoft® Technet, http://www.microsoft.com
[2] Microsoft® Windows® Server 2003 deployment kit, http://www.microsoft.com
[3] Mastering Windows® Server 2003, Mark Minasi , Christa Anderson, Michele Beveridge, C.A. Callahan and Lisa Justice , Sybex Computer Publishing
[4] Windows® Server System™ Reference Architecture, http://www.microsoft.com
[5] Windows® Server 2003 Active Directory Planning, Implementation and Maintenance, Anil Desai and James Chellis, Sybex Computer publishing
[6] Active Directory Primer-Omar Droubi, Kenton Gardinier, Rand Morimoto and Michael Noel , Sams Computer publishing
[7] Monitoring and Troubleshooting Active Directory Replication, http://www.windowsnetworking.com

Incremental Learning Algorithm for Speech Recognition

Hisham Darjazini
h.darjazini@uws.edu.au

Qi Cheng
q.cheng@uws.edu.au

Ranjith Liyana-Pathirana
r.liyanapathirana@uws.edu.au

University Of Western Sydney
School of Engineering
Locked Bag 1797 Penrith South DC NSW 1797
Australia

Abstract **- This paper presents an implementation of incremental learning neural networks algorithm for speech recognition. The algorithm has been investigated using the TIMIT speech samples and it has been shown to demonstrate high recognition accuracy.**

KEY WORDS
Incremental learning, speech recognition

I. INTRODUCTION

Incremental learning updates a recognizer using the information obtained from unknown input data (ID). It has the advantage of adapt to changing input without requiring time-consuming in training. Incremental learning algorithms were mostly designed and tested for pattern recognition applications (References [1], [8], [9] and [10]).

Speech signals change from speaker to speaker and from time to time even for the same speaker. Incremental learning is a suitable tool for speech recognition. Though incremental learning has been applied to speech enhancement in Ref. [3], very little research has been reported in the literature on its use in speech recognition.

In this paper, we propose an implementation to feed-forward incremental learning algorithm based on method developed by Darjazini and Tibbitts (Ref. [2]).

II. WEIGHT SET ADDITION ALGORITHM

The weight set addition algorithm is based on a method for speech recognition that employs a comb of phone sub-recognizers (Ref. [2]). As shown in Fig. 1, the method employs 55 sub-recognizers for recognition of 54 phones and silent period.

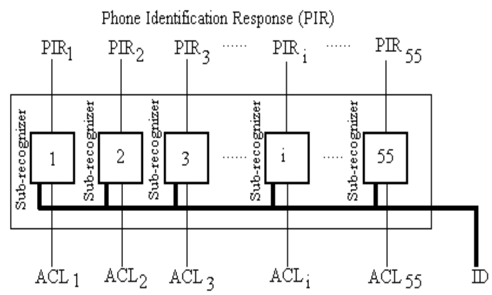


Fig. 1. Comb of Sub-recognizers.

All the sub-recognizers are implemented using an identical feed-forward neural network (FF-NN). Each sub-recognizer has an output referred to as Phone Identification Response (PIR), where the PIR is a continuous variable between 0 and 1. Each sub-recognizer indicates that the input speech contains a specific phone if the value of PIR is close to 1. The tolerance in the network is set to 0.05. Therefore, each PIR with a value of greater than or equal to 0.95 is taken as an indication of a potential match.

The algorithm extracts a new weight set (WS) from a new unknown data set during recognition. In this algorithm, the sub-recognizer contains two phases of back-propagation instead of one. At the initial trial the network behaves as a normal back-propagation network. At the subsequent trial the network performs the incremental learning process firstly by running the network using the available weight set, at this point a measure applies at the output, if the resulted error comes greater than the maximum allowed error, the process

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 231–235.

will be terminated as a nonrecognized phone. If the error is less

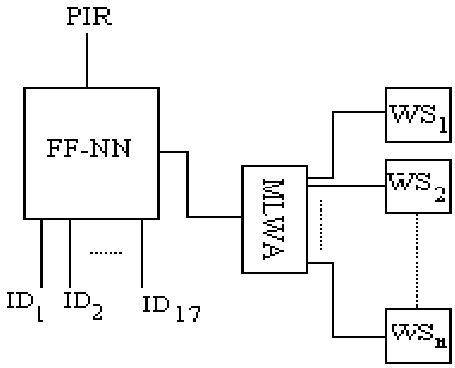


Fig. 2. Choice of Weight Set for Incremental Learning.

than the maximum allowed error and higher than a minimum acceptable output; then the incremental learning phase starts. The goal of the incremental learning phase here is to achieve an acceptable output at the output layer by adjusting the weight set using adaptive learning rate. When this is achieved the new weight set will be saved and sent to the MLWA (see Fig. 2) for later reference.

In the following recognition, the new set, as well as all the existing sets, will be tested as a potential weight set candidate in the FF-NN. The weight set that produces the highest PIR is selected as the most recent updated weight set. This function is performed by the Most Likelihood Weight Activator (MLWA) unit, which is shown in Fig. 2.

In Fig. 2, WS_1 is obtained from training session, obviously, in the early stages of incremental learning. Subsequent WS_S along with WS_1 will have statistical order in the MLWA and the highest probability WS is the one which is used mostly. Other sets will be used more often later on.

Fig. 3 shows the multi-layer structure of a sub-recognizer. The input layer contains 17 processing elements (PE) used to receive 17 input elements, which represents the Mel-scale Frequency Coefficients (MFCC) of the corresponding phone. In this structure, the input layer acts as a buffer to the subsequent hidden layers. There are three hidden layers H_1, H_2, and H_3, each one containing (34 - 51 - 34) PEs respectively. The output layer contains one PE representing a measure of the matching of the input speech (stimulus) to a particular phone.

III. Experimental results and discussion

The input data was extracted from 75 spoken sentences of the TIMIT speech database. The sentences are spoken by 25 speakers (5 female and 20 male). Every speaker posses one of three main dialects from the American English, and the accent was chosen arbitrarily. The data was mixed to produce as much variety as possible to every phone; this is to get the advantage of having the sub-recognizer being exposed and to deal with most varied forms of the same phone. In the primitive representation of the input data, there were 54 distinctive phones appeared in 2440 samples, which were segmented from 637 words. The table in the appendix shows these phones and their number of occurrence.

Experiments were performed firstly by initiating (first run) the sub-recognizers using the back-propagation learning algorithm and applying the Delta rule. The exit condition of this session was the number of iterations, which was set at 500, and the learning rates were all initiated to 0.5. The weights were initiated to random normally distributed values and the learning set contained nonclustered stimulus. Maximum accepted error (tolerance) is 0.01 and the incremental learning width is 0.219, i.e. the range is from 0.989 to 0.77

The initial session provides the first weight set (WS_1) for the MLWA and determines the first cluster of the input data. The number of phone samples for the initial session was in this case 15, and the sub-recognizer converged from the target after 50 epochs. In each epoch, the network manipulated the inner weights of the hidden layers. An error monitor was set to measure the value of mean squared error (MSE) value at each hidden layer and the effects of a particular PE on the overall result of the network. The accuracy of the PIR was within an error value of 0.01, which is below the tolerance value. The overall performance on the initial learning set scored 94.44% accuracy.

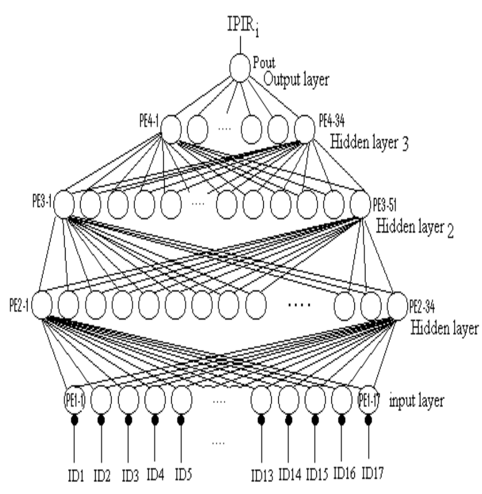


Fig. 3. Structure of Sub-recognizer.

Fig. 4 illustrates the performance of the sub-recognizer in the initial session, where Fig. 4(a) illustrates the measure of the mean square error (MSE) graph and Fig. 4(b) illustrates the PIR values at the end of the initial session. It can be noted that the network converged successfully within short time measured at about 50 epochs. The rest of the samples have been presented to the network in the incremental learning stage where the performance was close to 99.20%. The failed cases produced from samples with PIRs out of the predetermined incremental learning range.
In the initiation session, some of the phone sets required up to 13 trials to achieve convergence. This was partially due to the wide range of the used phone types in the input data. The diversity of the input data resulted in the wide distance between some of the stimuli presented to the network.

Fig. 5 illustrates examples of two trials on the phone / s /, where, Fig. 5(a) shows the MSE and the PIR for one of the nonconverged trials. When that occurred, the trial was restarted again (based on a new randomly generated weight set) and at the end a convergence was achieved as shown in Fig. 5(b).

In several phone sets used above, there was some overlapping in the phonemic borders. Therefore, amalgamation have happened to some phone sets and one set of phones was taken as a unique cluster in the larger phone set, which was used to create a larger learning set. For example, the set / b / and / bcl / were merged together. This is aimed to achieve larger variety of the phone forms in the phonemic knowledge representation. This amalgamation resulted in more learning sessions to run and more complex work to be performed at MLWA. Finally, the number of distinctive phones in the phonemic knowledge was chosen to be 55 and the overall performance of the system converged to the target.

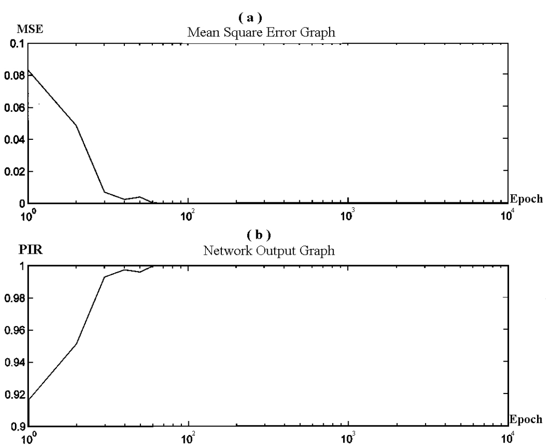


Fig. 4. The Sub-recognizer Performance in the Training Session.

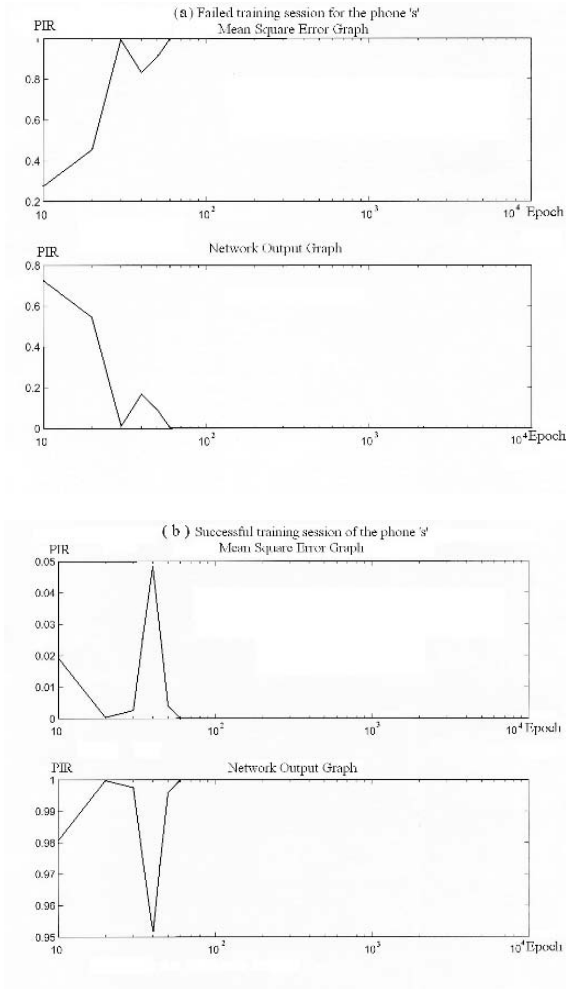


Fig. 5. Illustration of Training Experiments on the Phone / s /.

IV. CONCLUSION

The proposed incremental learning algorithm allows the original sub-recognizers to be updated without causing the system to lose its original phonemic knowledge or suffer from catastrophic forgetting problems. The system has demonstrated excellent performance.

One critical parameter worth mentioning here is that the incremental learning range is a very critical parameter for the system performance and has to be predetermined. The wrong range could result in false recognition in the phonemically adjacent phones and may lead to catastrophic forget situation.

ABBREVIATIONS

ACL = Activation Control Line.
FF-NN = Feed Forward Neural Networks.
ID = Input Data.
MFCC = Mel-scale Frequency Cepstrum Coefficients.
MLWA = Most Likelihood Weight Activator.
MSE = Mean Square Error.
PE = Processing Element.
PIR = Phone Identification Response.
WS = Weight Set.

APPENDIX

Phones set used in the learning trials and their relevant number

Phone	Number of samples
ch	12
jh	15
dh	48
f	33
s	126
sh	38
th	10
v	40
z	42
em	3
en	15
eng	1
m	73
n	137
ng	23
nx	13
epi	21
h#	1
pau	22
eI	18
hh	15
hv	24
L	82
r	87
w	43
y	24
b	43
bcl	3
d	59
dcl	13
dx	44
g	23
gcl	3
k	87
kcl	13
p	51
q	64
t	85
tcl	22
aa	64
ae	69
ah	34
ao	41
aw	15
ax	75
axh	7
axr	41
ay	31
eh	57
er	37
ey	46
ih	91
ix	136
iy	112
ow	38
oy	11
uh	9
uw	6
ux	28

REFERENCES

[1] D. Chakraborty and N. Pal, A novel learning scheme for multilayered perceptrons to realize proper generalization and incremental learning, *IEEE Transaction on Neural Networks*, 14(1), January 2003, 1-14.

[2] H. Darjazini and J. Tibbitts, The construction of phonemic knowledge using clustering methodology, *Proceedings of the Fifth Australian International Conference on Speech Science and Technology*, Dec. 1994, Vol. 1, 202-208.

[3] L. Deng, J. Droppo and A. Acero, Incremental Bayes learning with prior evolution for tracking nonstationary noise statistics from noisy speech data, *Proceedings of IEEE International Conference on Acoustic Speech and Signal Processing*, Apr. 2003,Vol. 1, 6-10.

[4] L. Fu, H. Hsu and J.C. Principe, Incremental backpropagation learning networks", *IEEE Transactions on Neural Networks*, 7(3), Nov. 1996, 757-761.

[5] C.M. Higgins and R.M. Goodman, Incremental learning with rule based neural networks, *Proceedings of IEEE International Joint Conference on Neural Networks*, 1991, 875-880.

This page intentionally blank

[6] T. Hoya, On the capability of accommodating new classes within probabilistic neural networks, *IEEE Transaction on Neural Networks,* 14(2), Mar 2003, 450-453

[7] W.L. Mahood, Incremental learning mechanisms for speech understanding, *Proceedings of the IEEE International Workshop on Tools for Artificial Intelligence, Architectures, Languages, and Algorithms*, Oct. 1989, 237-243.

[8] R. Polikar, L. Udpa, S. Udpa and V. Honavar, Learn++: An incremental learning algorithm for supervised neural networks, *IEEE Transaction on Systems, Man, and Cybernetics - Part C: Applications and Reviews,* 31(4), Nov. 2001, 497-508

[9] M.T. Vo, Incremental learning using the time delay neural network, *Proceedings of IEEE International Conference on Acoustic Speech and Signal Processing,* Vol. 2, April 1994, 629-632.

[10] D. Wang and B. Yuwono, Incremental learning of complex temporal patterns, *IEEE Transactions on Neural Networks*, 7(6), Nov. 1996, 1465-1481.

Manufacturing Process Modeling

Isidro Moctezuma-Cantorán[1], Manuel Valdés-Marrero[1], Jorge Ochoa-Somuano[1] and Isaías May-Canché[2]
(1) Universidad del Mar, Campus Puerto Escondido
Av. Universidad s/n, Puerto Escondido, Oaxaca, México
CP 71980, Tel. 52 + 954 588 3365
(2) Instituto Tecnológico de Chetumal
Av. Insurgentes #330, esq. Andrés Quintana Roo, Col. David Gustavo Gutiérrez, Chetumal, Quintana Roo, México
CP 77013, Tel. 52 + 983 832 1019 and 983 832 2330 Ext. 128

***Abstract*-This paper describes the development of a visual environment prototype that allows the manufacturing processes design and modeling using LAB (Besançon Automatics Laboratory) modeling. LAB models are used to perform design and modeling in four levels of detail, from abstract to concrete. To validate the relations within the model in a formal way, relations visual grammars are used. Finally, this paper describes an example of a manufacturing process.**

I. INTRODUCTION

In manufacturing processes, one of the stages that has been neglected and without a doubt is very important is the planning and design of the processes required to elaborate a product as well as the required equipment. In the following sections of this paper, LAB models are described, giving the definition of each model and the description of each level of detail. Once all models are defined, relations visual grammars are used to formalize the relations between models. Finally, a modeling example is presented.

II. LAB MODELING

The LAB (Laboratoire d'Automatique de Besançon) in France developed LAB modeling [1]; with the purpose of contribute to the complete and integrated conceptualization of manufacturing processes in four levels of abstraction, specifying in a progressive manner the product, the process tools and the necessary equipment. This systematic analysis approach allows complexity reduction in the generated process descriptions, helping in their comprehension by different people involved in the manufacturing process.

In this work context, the fabrication universe is composed by three base sets of elements that are involved in a manufacturing process. The three sets are:

1) A set named P, and its elements are the basic components of a product.

2) A set named H, which consists of the tools necessary for the transformation of P components. These tools can be a *palett* (for containing or transporting a product component), a *gripper* (for holding and manipulating), or cutting tools (such as grinders, drills, saws, etc.), among other tools.

3) A set named M, which consists of the means to generate mechanical energy or otherwise and transformation or process tool (H elements) bearers. This set can contain the following elements: robot arms, CNC machines, motors, conveyor belts, and other equipment.

The tasks performed in the manufacturing of a product are described by [2] in four levels of abstraction:

Level 1: Functional Scheme. With the use of functions, it describes the integration of components with the purpose of obtaining a product.

Level 2: Operational Scheme. It describes the operations performed on product components in a given location and with a given orientation on the installation space. It considers only product components without the equipment and the means to elaborate them.

Level 3: Execution Scheme. The operation describes product components and transformation tools (H set) involved in the processes required for manufacturing a product.

Level 4: Action Scheme. The actions are sets of product components, transformation tools and bearers of these transformation tools (M set). Actions include tasks represented in the previous three levels plus the means necessary (e.g. a robot arm, a conveyor belt, etc.) for the manufacturing of a product.

An example is shown in Fig. 1.

Note: The difference between the elements of sets H and M is that elements in H act directly upon the product components and elements in M act using an element in H.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 237–242.

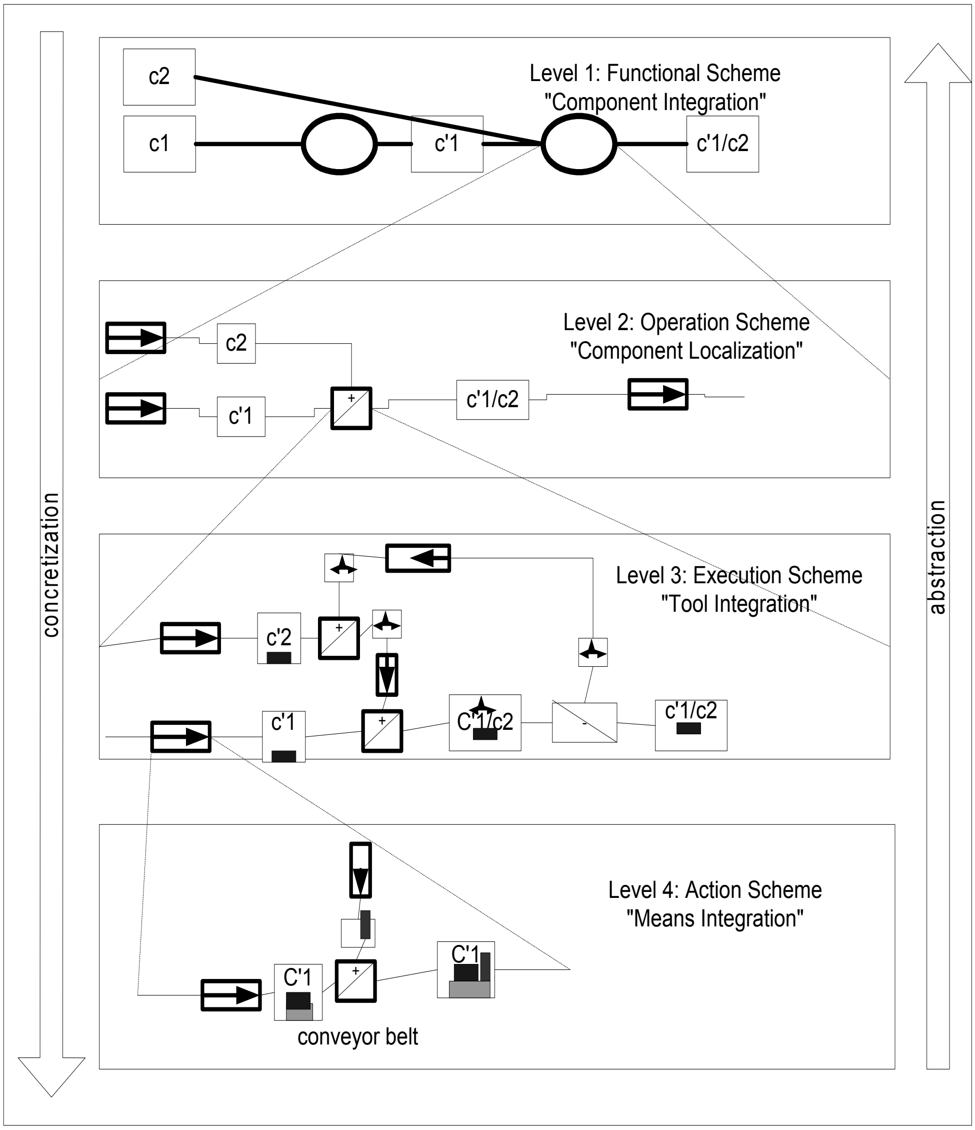


Fig. 1. Detail description of a task using four levels of detail.

III. Symbolic Relations Grammar

Once the model is defined, a formal Symbolic Relations Grammar (GRS) is applied to generate a visual language as is specified by [3] and [4]. In a GRS a sentence is seen as a set of consecutive symbols.

Definition: A GRS is part of the visual grammars and these are part of the Context Free Grammars. Is a sextuple G = (N, T, I, R, Atr., P) where:
N: Is a finite set of non-terminal symbols.
T: Is a finite set of terminal symbols.
I: Begin symbol.
R: Is a finite set of relations symbols.
Atr.: Is a finite set of attributes symbols.

P: Is a finite set of productions.

Defining our grammar for the visual symbols on level 1: "Component Integration," we have G = (N, T, I, R, Atr, P.).

N = {Diagram}

T = { physical operation, ensemble, disassemble, component}

I = {Diagram}

R = {Next}

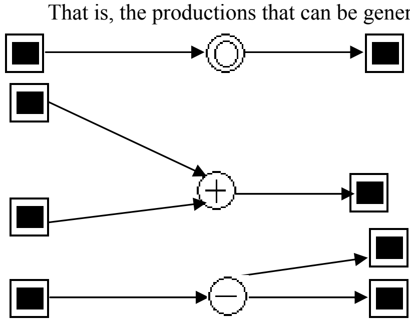
Atr.= {input, output}

P = {Diagram ==> 1, 2, 3, Next (1 (input 3)) & Next ((output 3) 2).

Diagram ==> 1, 2, 3, 4, Next (1 (input 4)) & Next (2 (input 4)) & Next ((output 4) 3).

Diagram ==> 1, 2, 3, 4, Next (1 (input 4)) & Next ((output 4) 2) & Next ((output 4) 3). }

That is, the productions that can be generated are like:



The grammar for level 2: "Component Localization" has the visual elements that represent the different kinds of operations, also it has the input and output objects which are . This visual element represents a product component. So, we have G = (N, T, I, R, A, P), where:

G = (N, T, I, R, A, P).

N = {Diagram}

T = { displace, disassemble, ensemble, physical operation, component}

I = {Diagram}

R = {Next}

A = {input, output}

P = {Diagram ==> 1, 2, 3, Next (1 (input 3)) & Next ((output 3) 2).

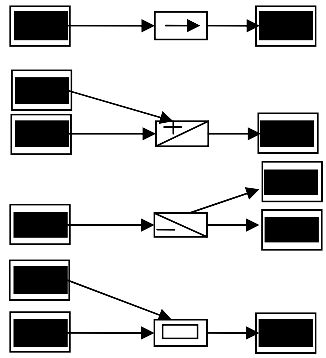
Diagram ==> 1, 2, 3, 4, Next (1 (input 4)) & Next (2 (input 4)) & Next ((output 4) 3).

Diagram ==> 1, 2, 3, 4, Next (1 (input 4)) & Next ((output 4) 2) & Next ((output 4) 3).

Diagram ==> 1, 2, 3, Next (1 (input 3)) & Next ((output 3) 2).

}

That is, the productions that can be generated are like:



Similarly, grammars for levels 3 and 4 are generated.

IV. Results

As a result of applying visual grammars, we developed a software tool, and it is in the prototype phase. This tool allows the modeling of manufacturing processes that can be automatic or semiautomatic, and it produces diagrams which are lexical and syntactically correct and complete [5].

Now, we present the modeling of the necessary tasks and processes required for the bottling of an alcoholic beverage. The bottling process consists of the following tasks:

1. Washing of the bottle, with pressurized water (mechanically).
2. Filling the bottle (mechanically).
3. Placement of caps (mechanically).
4. Quality control (manually).
5. Placement of labels (manually).
6. Placement of quality seal (manually).
7. Insertion of the bottle in the heat tunnel (mechanically).

Note: The displacement of the bottle in the production line is performed with a conveyor belt.

Note: Manual activities are those tasks which are performed by a human with physical effort; mechanical activities are those tasks which are performed by mechanical, semiautomatic or automatic equipment. These also apply to the other models.

In Level 1: "Component Integration" modeling, the necessary components and functions (tasks of manufacturing processes) for obtaining a bottled product are specified. In this case, the components are:

- Bottle.
- Liquid.
- Cap.

- Label.
- Quality Seal.

The necessary tasks are shown in Fig. 2.

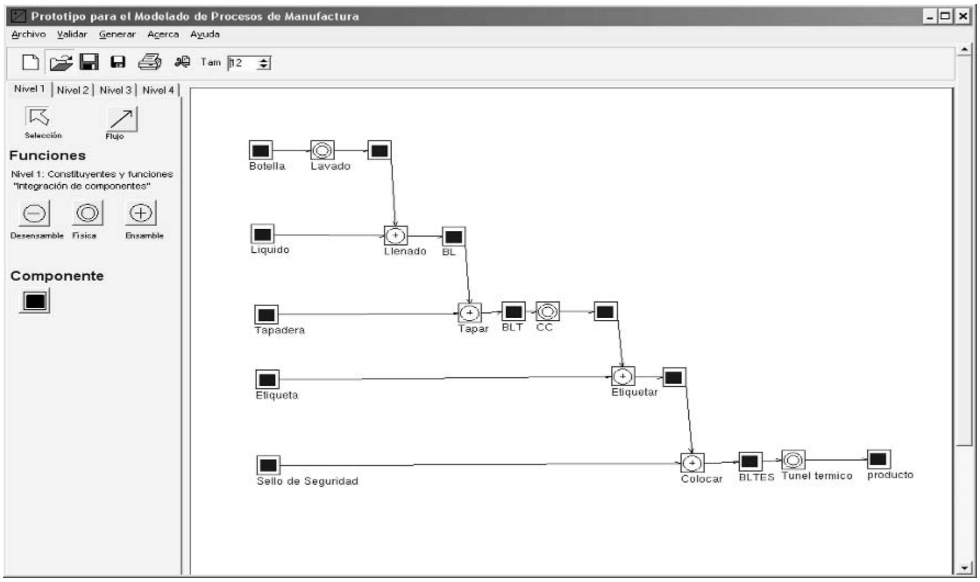


Fig. 2. Level 1: "Component Integration."

In level 2: "Component Localization" modeling, the localization of tasks within a manufacturing installation is specified, as shown in Fig. 3. It should be noted the addition of the displacement visual element.

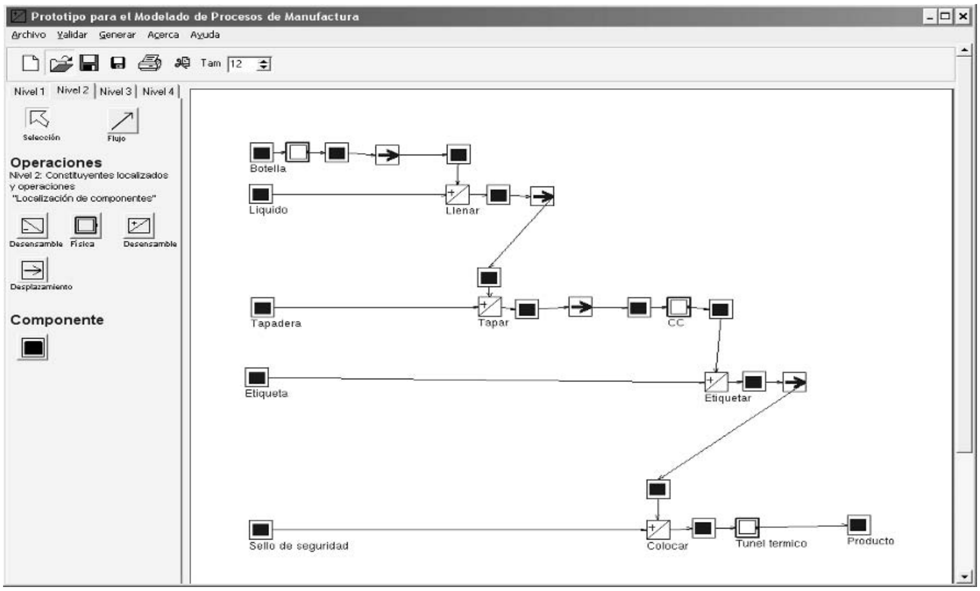


Fig. 3. Level 2: "Component Localization."

In level 3: "Tools integration" modeling, necessary tools to perform the given tasks are added, as shown in Fig. 4. It should be noted the addition of different kinds of visual elements, which represent grippers, machines, manual work, and other tools.

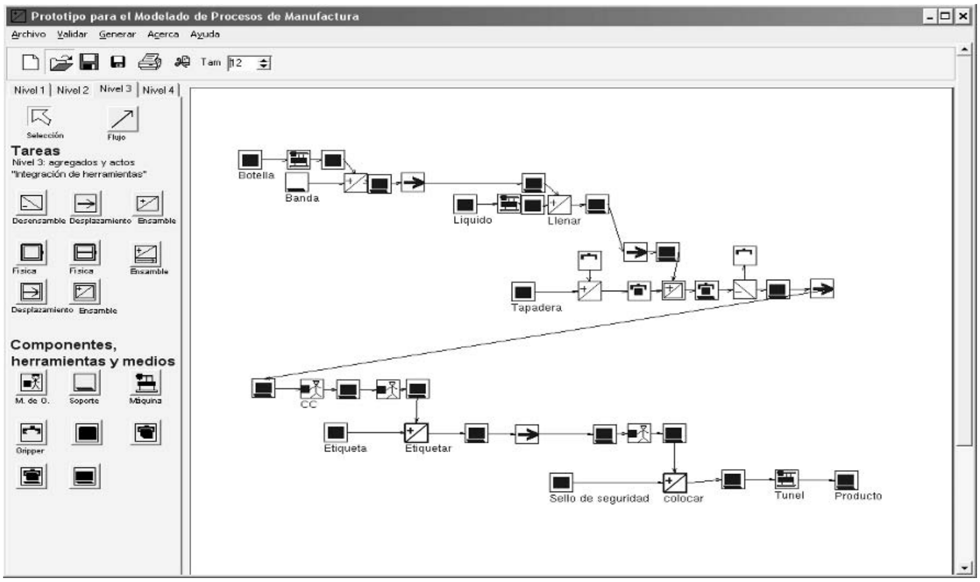


Fig. 4. Level 3: "Tools Integration."

In level 4: "Means Integration" modeling, the necessary means for moving the tools as well as for performing the tasks are added. This is shown in Fig. 5.

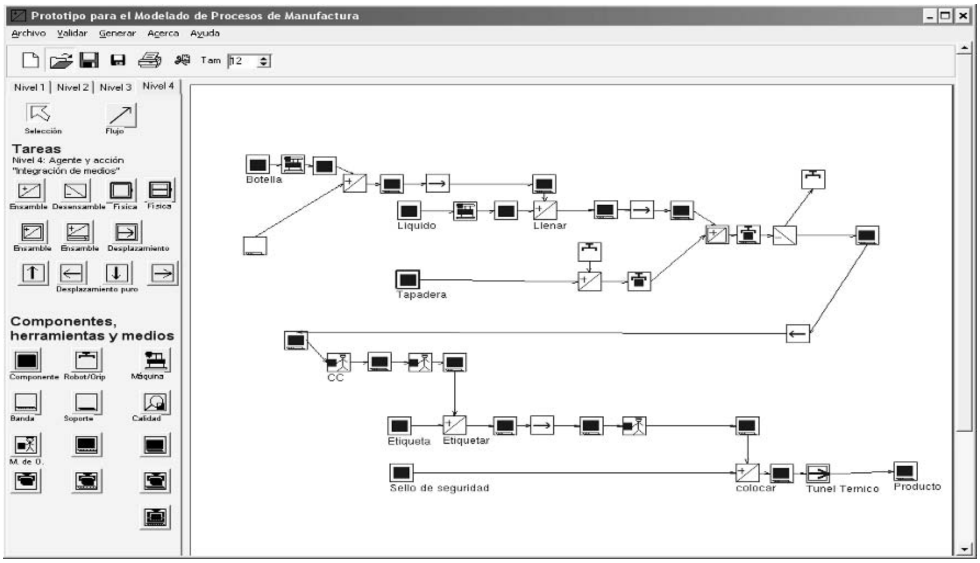


Fig. 5. Level 4: "Means Integration."

V. Conclusions

We developed our visual environment software tool prototype to ease the tasks of designing and modeling manufacturing processes. This tool uses a unified modeling technique called LAB methodology. This technique allows the integrated conceptualization of manufacturing processes in four different levels of abstraction, adding in a progressive manner the product components, the process tools and the equipment.

With this product, product engineering or method engineering staff, as well as production personnel can communicate and establish design strategies for the product, the installations, and the planning of production, in accordance with the different abstraction levels of the manufacturing tasks.

The tool may also be an aid in the educational sector since it facilitates the learning of manufacturing systems in manufacturing or engineering courses.

A. Future Work

Several future works that may complement this research are the following:

- Incorporate the grouping of visual elements, allowing the design of bigger schemes, because sometimes in reality the manufacturing process will not fit the screen or are to big to understand.
- Incorporate the simulation of processes, the calculation of times, costs and critical route.

References

[1] A. Zimmermann, A. Kühnel and G. Hommel, “A modeling and analysis method for manufacturing systems based on Petri-nets,” *In: Proc. Computational Engineering in Systems Applications (CESA’98)*, 1998, pp. 276-281.

[2] O. Barakat, *Contribution à la modélisation et à la simulation orientées objet des systèmes flexibles de production*, Masters Thesis, Franche-Comté University, France, 1991.

[3] K. Marrit and B. Meyer, *Visual Language Theory,* Springer-Verlag, 1998.

[4] R. Johnsonbaugh, *Discrete Mathematics*, Prentice Hall, 1997.

[5] I. Moctezuma, *Ambiente visual para el diseño y modelado de procesos de manufactura*, Masters Thesis, National Centre for Research and Technologic Development, Mexico, 2004.

Architecture for Virtualization in Data Warehouse

J. A. Nasir *M. Khurram Shahzad*
janasir@pucit.edu.pk *mkshahzad@ciitlahore.edu.pk*

Punjab University College of Information Technology,
University of the Punjab, Lahore.

ABSTRACT

Conventional data warehouse (DW) due to structure of its schema and contents is unable to: a) support any dynamics in its source structure and contents b) unable to support hidden-subjects c) unable to provide data on-the-fly i.e. real-time data and populate hidden-subjects on their evolution. To handle these problems the concept of virtualization in DW is floated here.

In this study we have proposed architecture of virtualization approach. According to this approach, conventional DW is replaced by: i) a storage component called data-store ii) a Synthetic warehouse (SWH). Data-store is a non-subjective, content consistent, time-variant and integrated storage. On the other hand, SWH is only a structure, with no instances attached. It acts as a schema source for analytical processing and is mapped to its data-store. Subjective conversion is expected to be done on-the-fly. We are hopeful that this architecture will qualify all the evaluation parameters of: i) scalability ii) hidden-subjective support iii) source dynamics.

KEYWORDS

Data Warehouse, virtualization, on-the-fly integration, architecture, synthetic warehouse.

1. INTRODUCTION

Data warehouse (DW) integrates these various operational sources and depends upon operational sources for all changes. Changes in operational sources may result in derivation of in-consistent results [2, 3, 4]. These changes can be, schema changes or data changes. A number of efforts [3, 4, 5] have been made by the researchers of the domain to handle these changes.

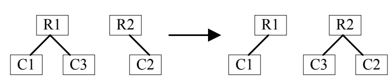
Observations-- It is observed that above approaches, due to structure of its schema and its contents are unable to: a) Support hidden-subjects b) Unable to provide real-time data availability, its importance can be found in [8] c) Cannot handle source dynamics.

Contribution-- Our approach in this paper to address the issues is based on Synthetic Warehouse (SWH). Non-subjective data from multiple sources is stored in a central repository after integration, cleaning and removing in-consistency. The repository is to be called Data Source (DS). Integration, cleaning and inconsistency removal is done by a component called Real-time ETL (R-ETL). To meet real-time what-if analysis requirements data is transformed from DS by mapping through SWH.

2. PROBLEM DEFINITION

DW describes real world, by providing integrated and subjective access to operational sources that are likely to change with time. These changes may result in production of obsolete results, so DW don't meet dynamic users requirement for decision support [3]. Changes to these sources has been categorized many times into two types [4, 7, 8] '(i) Content Changes (ii) Schema Changes. Multiple attempts have been made in [3, 5, 8] but are not successful due to one reason or the other. The problems addressed in this paper are:

1) At times, changes to DW schema are required to propagate content changes of operational sources as given in example by Eder [3]. The example is: consider a company, with sale points in multiple regions. Each region has multiple cities, if boundaries of region are changed which results in transferring the city from one region to another as shown in fig.1. For these changes, obsolete results are produced.



Rj represents Region with ID j
Ck represents City with ID k1

Fig. 1. Changing Region of City [9]

2) Conventional DW does not support subjective-scalability, i.e. hidden subjects cannot be catered by conventional subject-oriented data schema in DW. Usually, data from sources is transformed into subjective structure of DW after removing inconsistencies and

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 243–248.

anomalies; also operational data sources are either dissolved or off-lined after a specific time. In such case if subject is evolved, it is not possible to populate new subjective-schema, due to the absence of operational sources'.

3) Real-time integration is not provided in conventional data warehousing techniques. But recent study [6] has identified that real-time data is required in DW for better performance and optimization of effective decision-making in DW, which is not maintained in the conventional DW.

3. EXISTING APPROACHES

According to the best of our knowledge, first approach to handle changes to the source was, isolate changes of schema, this can be accomplished with the help of a middleware. Schema evaluation [5] was another approach. This approach supports only one version, but, regular up-gradation of DW schema, transformation package and multi-dimensional structures increases the maintenance cost. Third approach to handle the issue of source dynamics is versioning approach [10], in which changes are applied to new version and both versions are maintained.

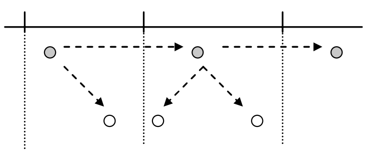


Fig. 2. Schema Versioning Graph [11]

According to recent version approach [10], changes to schema are applied to new version called 'Parent Version' which is explicitly derived from 'Child Version'. Schema versioning graph is used for demarcating parent and child relationship, as shown in fig.2. All these approaches may some-how manage source dynamics but doesn't support on-the-fly refresh facility for DW as well as subjective scalability.

4. VIRTULIZATION APPROACH

Our approach to handle above mentioned problems is based on virtualization of data warehouse. This approach avoids the evolution problem of 'hidden- subjects' by maintaining selected time-variant dataset. This data set acts as a source, for Synthetic Warehouse, for derivation of dimensions and most importantly facts.

Due to easy and costless availability of increased amount of space it is not a problem to maintain tera-bytes of data for decision-making. So for such purpose the conception of data-source (DS) has been floated which stores time-variant datasets in its non-subjective schema. For such purpose real-time transformation is done by component called R-ETL. It integrated data on the fly from operational data-sources for DS.

OLAP being the information delivery component refreshes data from DS by using Synthetic data warehouse (S-WH). S-WH is a virtual data warehouse with subject-oriented, time-variant logical structure. The source to S-WH is the data-store. Data is propagated to OLAP by mapping S-DW to DS.

4.1. ARCHITURAL COMPONENTS

Typical architecture of virtualization in DW is shown in figure 3. It includes tools for extracting data from multiple operational databases and external sources; for cleaning, transforming and integrating this data; for loading real-time data into the DS; and for periodically refreshing DS to reflect updates at the sources and to purge data from DS by S-DW to be used for OLAP by rule-mapping. The major components are:

4.1.1. *Data Sources*

Also called operational source systems and transaction processing systems. These systems keep data for a particular application [1] like order-processing, stock status, customer retentions etc. The left side of figure 3 shows set of heterogeneous transactional sources, which are used to meet the transactional requirements of the organization. It is to be note that these sources can be relational, object-oriented data sources, legacy systems and web sources. Heterogeneity in these sources can broadly be categorized into two types: Schema heterogeneity, content heterogeneity.

4.1.2. *Real-time Extractor, Transformer and Loader (R-ETL):*

Transactional sources act as data sources for subjective data [1]. For precise decision-making it is required to provide end users with most recently available data called on-the-fly data [6]. R-ETL performs integration tasks, real-time extraction and transformation and loading operations. Loading is done into a new database called data-store. Once data is loaded, it can further be used for various analytical purposes. Loading process must have following properties: a) Real-time loading b) Scheduled loading c) Consistent loading

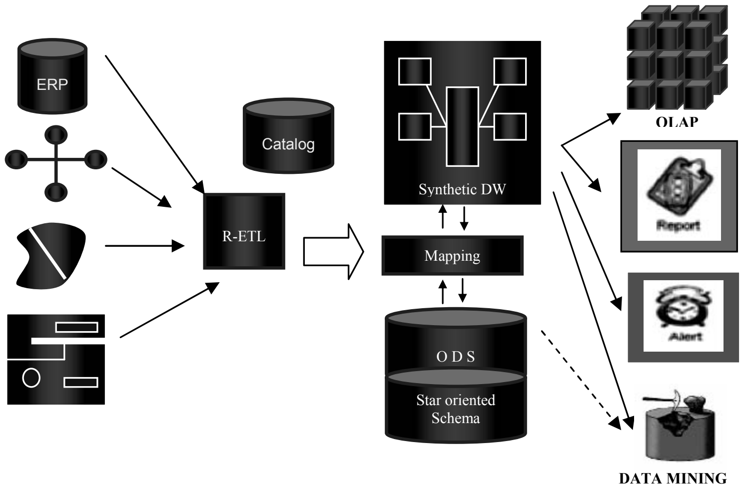


Fig. 3. Architecture for Virtualization in Data Warehouse

4.1.3. *Data Store (DS):*

Data store has been defined in various ways, but for virtualization architecture data-store is defined as:

Definition 1 – Data store

A data-store is a content consistent, real-time, time-variant collection of integrated operational data that will act as a source for Synthetic-warehouse.

This component plays major role by maintaining real-time integrated data from various operational data sources, which is done by R-ETL. DS with its time-variant schema maintains operational data, which can act as a source for dimensions and facts. Since all the possible derivates can be acquired from DS's schema, the schema adjustments can be avoided in the presence of contents changes in operational sources.

Our approach, in this study is based on on-the-fly subjective conversions, because it makes no sense to store summary data in the DS. Some other reasons are: a) Data stored in DS is constantly changing, b) may subject to schema and content changes, c) it is useless to store summary data in the presence of new computing age [12, 16].

4.1.4. *Synthetic Warehouse (SWH)*

Proposed architecture don't contain conventional data warehouse, as its component, instead it has a virtual data warehouse named Synthetic warehouse. SWH is defined as:

Definition 2 – Synthetic warehouse

Synthetic warehouse is a subject-oriented, time-variant virtual (instance-less) warehouse in favor of decision-support.

SWH with star-oriented structure (called dimensional schema) feeds OLAP and other analytical applications, by mapping DS to provide real-time analysis for end-users. Like conventional data warehouse, SWH contains two types of tables: i) Dimensions Tables ii) Fact Tables. During analysis, facts are analyzed in terms of dimensions which make the schema to subject-oriented form [1]. Unlike, conventional DW, SWH has only logical structure, with no instance; this logical structure is mapped to data store.

SWH is a federated solution; it maintains data in its DS rather materializing SWH. A separate component 'mapping manager' grabs requisite data, applies necessary transformation and deliver to the requester (OLAP). The repository of the Mapping-rules (mappings), available in catalog, helps out source data (i.e. DS data) for subjective format conversion.

Mappings are the set of rules, which are used to modify query for execution on DS. A query-processor with its operational-confinement layer will divide input query into small queries for populating OLAP cubes by using

SWH catalog. Scalability is one of the inherent capabilities of SWH, as the virtual pool of subjects.

Synthetic Warehouse Construction

Partial orders naturally facilitate multiple data organizations. Subject-oriented systems tend to encourage modules to identify unique and fixed classification of data, because the type of schema is usually taken for one of its permanent characteristic [13]. Synthetic warehouse acts as a subjective-schema- source for data driven intelligent applications. All hidden subjects are only catered with partial ordering. The mathematical partial ordering will support the construction of the SWH schemas. The first construction in the form is a subjective reduction, from objective sources or schemas. The root element U, set of all schemas, and the descendants $O_1, O_2, \ldots, O_n$, will be any demanded collection of schemas maintaining the partial order relationship with U. Figure 4 shows the construction of U.

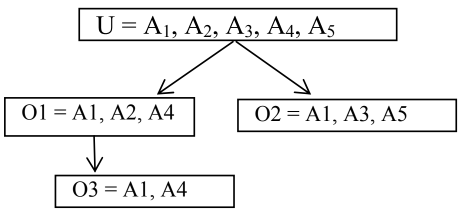


Fig. 4. SWH construction by partial order

Here A = $\{U, O_1, O_2, O_3\}$ is the base set. We can adopt the notion $(A, \subseteq)$ for this partial order, where U is maximal of partial order. The partial order explicitly separates the ordering relation from the base set. Standard and renown mathematical representation of this partial order is.

Reflexive:

Clearly for any $O_i \in A$, $O_i \subseteq O_i$.
So $\subseteq$ is reflexive in A.

Anti-symmetric

For any O_i and $O_j \in A$, $i \neq j$.
If $O_i \subseteq O_j$ then $O_j \not\subset O_i$.

Transitive

If $O_i \subseteq O_j$ and $O_j \subseteq O_k$
then $O_i \subseteq O_k$

Hence $\subseteq$ is a partial order in A.

Construction Algorithm

The construction algorithm to be used is:

```
Select (Oi from A)
{Search Oi from A
      If Oi is found in A
    Then Give results;
  Else
Find appropriate parent P of Oi
   If Parent P found
  Add Oi as a child of P
  Else
  Add Oi as a new node;
  End
  End
  }
```

Harmonizing n DS for SWH

SWH has full potential to be mapped to multiple data-stores (DS), which may require to be harmonized. Here, for harmonizing SWH over multiple DSs, we have taken Genesereth's technique [14] without any modification, which are discussed in two cases:

Case 1: Schema integration

Consider multiple DS_1 and DS_2 be two data-stores. DS_1 have table T_1 with attributes $\{A_{11}, A_{12}, A_{13}, M_1\}$ and DS_2 have T_2 with attributes $\{A_{21}, A_{22}, A_{23}, F_2\}$. Both these can be matched to virtual dimension T of SWH. Attributes of T then becomes $\{A_1, A_2, A_3, M, F\}$. The following rule shows the attribute mapping from DS_1 T_1 to Virtual dimension T of SWH.

((<= (A_1 ?x ? y) (A_{11} ?x ?y) =>)
(<= (A_2 ?x ? y) (A_{12} ?x ?y) =>)
(<= (A_3 ?x ? y) (A_{13} ?x ?y) =>)
(<= (M ?x ? y) (M_1 ?x ?y) =>))

The following rule shows the attribute mapping from DS2 T to Virtual dimension T of SWH.

((<= (A_1 ?x ? y) (A_{21} ?x ?y) =>)
(<= (A_2 ?x ? y) (A_{22} ?x ?y) =>)
(<= (A_3 ?x ? y) (A_{23} ?x ?y) =>)
(<= (F ?x ? y) (F_2 ?x ?y) =>))

[14] has given the above mapping rules for only schema integration with out any content change.

Case 2: Content integration

Similarly, content integration has also been given by Genesereth. Consider two DS_1 & DS_2, T_1 of DS_1 has $\{A_{11}, A_{12}, A_{13}\}$ and T_2 of DS_2 has $\{A_{21}, A_{22}, A_{23}\}$. Both these are to be mapped to T_1 of SWH with attributes $\{A_1, A_2, A_3\}$. Also consider A_{13} contain values in rupees, while A_{23} contain values in dollar. The mapping of A_1 and A_2 will remain same i.e. with out conversion, while the change rule for DS_1 is:

(<= (A_3 ?x ? y) (A_{13} ?x ?y) =>)

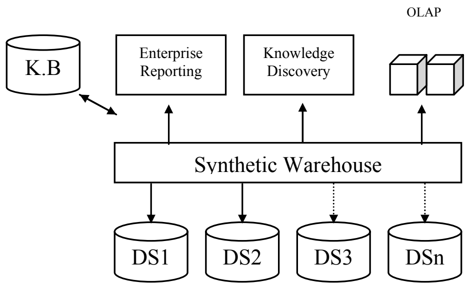


Fig. 5. Mapping of SWH to multiple DS

The change rules for DS

The change rules for DS_2 are:

(<= (A_3 ?x ? y) (A_{23} ?x ?dollar)
(round ?dollar rate ?y) =>)

This is the conversion of attributes of different values to same one. Figure 5 shows this mapping for ease of understandability

4.1.5.Information Delivery Components:

Interaction with the system is done by different paradigms. These can be: (i) OLAP a multidimensional structure used for what-if analysis [1]. Data storage to the multidimensional structures is done by mapping SWH to DS. (ii) Reporting [6], is done by using the DS/SWH, in which real-time integrated data is placed. (iii) Also, alerts and dashboards [6] are information delivery component by taking SWH as source. (iv) For predicting purposes Data Mining is used [15]. In the proposed architecture SWH acts as a source to it, if DS is used as source (shown in fig.3. by dotted arrow), predictive modeling becomes more effective due to increased number of considerations.

5. STRENGHTS OF VIRTULIZATION

Here we have proposed DW architecture based on the conception of virtualization using SWH, with no instance. Also, it provides a reliable support for decision-support. Virtualization strengths include:

a. *Content changes are accommodated:* To accommodate contents [4], proposes multiple versions. But not is the case with virtualization, DS with its schema has full potential of accommodating the content changes.
b. *Way-out to schema up-gradation:* By using DS usually it will not be required to up-grade schema. So schema modifications are not required, nevertheless any major change to operational sources results in schema modifications.
c. *Real-time integration:* Due to the presence of R-ETL, most recent data is transformed from the mirror to the DS which is further used for what-if analysis but after passing through S-DW. Making the analysis up to date and reliable.
d. *DW scalability for hidden subjects:* It is also mentionable that with the passage of time, subjects can also emerge. But conventional approach does not support hidden subjects, while in the proposed approach all possible hidden subjects can be populated by DS.
e. *Reliable prediction:* Mining by recognizing patterns usually provide sound founds for prediction [16]. Its probability of making prediction becomes more precise as soon as the real time source is provided to it. By using real-time integration DS provides a solid foundation to the predictive modeling.
f. *Reduced versioning:* To accommodate schema and content changes, it was proposed to maintain multiple versions, but with the passage of time writing query and retrieval becomes more complex. Proposed approach also reduces probability of increasing versions.

6. TECHNOLOGY CHALLENGES AND RESEARCH ISSUES

An architecture sketch for complete DW virtualization is proposed in figure 3. Left side of figure shows set of possible data sources e.g. these sources can be relation, object-oriented, legacy systems or Web sources. R-ETL tool extracts data from various operational data sources and cleans/transforms/integrates them and loads it into DS, which can further be used for various purposes. S-DW acts as a source for OLAP applications by mapping DS.

As a part of our project, initially we have only proposed virtualization architecture. Further in our project we are looking forward to solve number technical challenges which are discussed in this section. The issues are:

- In most cases it requires due consideration to adopt on-the-fly techniques to load DS, which raises serious problems in terms of data quality and integration.
- Most of the cleaning techniques devised so far purge problem [18] and duplication detection [17] rely on the presence of a materialized integrated level we

expect that, in its absence, some of these techniques can be modified to be re-implemented on proper data structures in main memory while other cannot be applied.

- Building a real-time ETL is an interesting challenge related to DS population, whose responsibility is to integrate and transform consistent data for decision-making.
- Dropping data by mapping queries to DS is also a challenging task.
- Schema definition, its advantages and disadvantages for DS.
- OLAP refreshment that is query generation and qualification for SWH.
- Query confinement for execution on DS and results integration to refresh OLAP.
- Amendments to mining techniques to use DS as data source
- Mapping-rule definition to populate OLAP.
- Maintaining data consistency and atomicity.
- A framework to handle hidden subjects

All these problems require due consideration, to implement reliable decision support system.

7. CONCLUSION

Commercial DW systems existing in market due to structure of their schemas and relationships between data cannot support the dynamic user requirements, real-time data integration and content changes without schema modifications. So to handle such problem we have proposed Synthetic-Data Warehouse, with zero instances. Instances are generated by mapping DS to populate OLAP.

8. REFERENCES

[1] Paulraj Ponniah. *Data Warehousing Fundamentals, A Comprehensive Guide for IT Professional.* Wiley Publishers, 1998. ISBN: 0471412546

[2] Jarke, M. Lenzerini, M. Vallilious, Y. Vassiliadis. *Fundamentals of Data Warehouses.* Springer-Verlag,, 2000.

[3] Bebel, B. Eder, J. Koncilia, C. Morzy, T. Wrembel, R. *Creation and Management of Versions in Multiversion Data Warehouse.* Proceeding of , 2004.

[4] Sjoberg, D. *Quantifying Schema Evolution.* Software Technology 35, 1, Page 35-54, 1993.

[5] Blaschaka, M.Sapia, C.Hofling, G.: *On Schema Evolution in Multidimensional Databases.* Proceedings of DaWak99, Italy, 1999.

[6] Golfarelli, M. Rizzi, S. Cella, I. *Beyond Data Warehousing: What's Next in Business Intelligence?* Proceedings of 7th International Workshop on Data Warehousing & OLAP, 2004

[7] Rundensteiner, E. Koeller, A. Zhang, X. *Maintaining Data Warehouses over Changing Information Sources.* Communication of the ACM, Vol. 3, No 6, 2000.

[8] Pasha,M.A. Nasir,J.A, Shahzad,M.K *Semi-Star Modeling Schema for managing Data Warehouse Consistency*. Proceedings of IEEE-ACM-NCET, Pakistan, 2004.

[9] Eder, J. Koncilia, C. *Changes of Dimension Data in Temporal Data Warehouses*. Proceedings of DaWak, 2001.

[10] Chamoni, P. Stock, S. *Temporal Structures in Data Warehousing.* Proceedings of the Data Warehousing and Knowledge Discovery DaWak99.

[11] Morzy T., Wrembel R.: *On Querying Versions of Multi-version Data Warehouse*. In Proceedings of 7th ACM DOLAP'04, Washington, USA, 2004

[12] Bernadrino, J. Madera, H. *Data Warehousing and OLAP: Improving Query Performance Using Distributed Computing*, Proceedings of CAiSE'00, Sweden, 2000.

[13] Raymond,D. *Partial Order Databases,* PhD Thesis, submitted to Department of Computer Science, University of Waterloo.

[14] Genesereth,R.M. Keller,A.M. Duschka,O.M. *Infomaster: An Information Integration* System. Proceedings of ACM-SIGMOD'97, AZ, USA

[15] Han,J. Kamber,M. *Data mining: concepts and techniques,* Morgan Kaufmann publishers, 2000.

[16] Inmon, B. *The Operational Data Store.* InfoDB, February, 1995.

[17] Baina, K. Tata, S. Bema;o, K. *A Model for Process Service Inteaction.* Proceedings of 1st Conference on Business Process management. Netherlands, 2003

[18] Monge, A. Elkan, C. *The field matching Problem: Algorithms and Applications.* Proceedings of KDD-96.

A Study of Software Protection Techniques

Jan M. Memon Asma Khan Amber Baig Asadullah Shah
Department of Computer Science
Isra University
Hyderabad, Pakistan
{janmohd, asmakhan, amberbaig, asadullah}@isra.edu.pk

Abstract- **Software piracy and tampering is a well known threat the world is faced with. There have been a lot of attempts to protect software from reverse engineering and tampering. It appears as if there is an ongoing war between software developers and crackers, both parties want to get an upper hand over each other as the time passes. Some of the ample techniques of software protection are reviewed, including multi-block hashing scheme, hardware based solutions, checksums, obfuscation, guards, software aging, cryptographic techniques and watermarking. All of these techniques play their parts imparted on them to protect the software from malicious attacks.**

Keywords: *reverse engineering, software protection, piracy, obfuscation, software aging, watermarking, checksums*

I. INTRODUCTION

Piracy of software has always been one of the burning threats to the software industry [33, 34, 35], especially after the advent of the Internet and the availability of many software analysis tools. Computer software is an important asset to any organization developing it as massive investment of time; money and intellectual capital are involved in its production. However, once produced, software is at risk to theft and misuse. It is estimated that tens of billion dollars of revenue is lost by the software industry due to software piracy alone. Piracy is no longer the only issue, but software tampering with the malicious intent of planting a Trojan horse in the end user's system is a worrisome possibility.

Today's complex software is of much value to its creator, whether that be a company with many products, or the only product of a small company. Software piracy is a major trade and industry problem. Great losses to software producers are due to unauthorized use and distribution of their products. Of much concern is the protection of the software, such that it will always retain the functionality which its creators intended, always protect the intellectual belongings embedded in the program, and spoil attempts to make illegitimate copies of the program. Much has been done to thwart network originated attacks, but little has been done to thwart hardware and software based attacks on the intellectual property embedded within a program. These attacks include modifications to a program to omit critical checks, such as license file checks, or reverse engineering of a key piece of a program's functionality.

Reverse engineering can also leads to code-lifting that consists of reusing a crucial part of a code, without necessarily understanding the internals of how it works, in some other software. These attacks are potentially very costly to the original software developer as they allow a competitor to abolish the developer's competitive edge by rapidly filling a technology gap through insights gleaned from examining the software.

In this paper, some of the approaches are discussed which are employed by software developers to provide ample software protection. In the next section, Multi-block hashing scheme is discussed. In section 3 some of the widely used hardware based approaches are discussed followed by checksums in section 4. One of the most promising software protection techniques, obfuscation is discussed in section 5, followed by Guards and Software Aging in sections 6 and 7, respectively. In section 8, some of the cryptographic techniques are discussed for protecting Java programs from reverse engineering along with some watermarking techniques in section 9. Comparison of these techniques is made in section 10 and finally, the conclusions are drawn in section 11.

II. MULTI-BLOCK HASHING SCHEME

In this scheme, a binary program is divided into several different sized independent blocks; each block does not contain any branching instruction to the next block. The blocks are encrypted with the hash value of the previous block. The last block is encrypted with the hash value of the second last block; the second last block is encrypted with the hash value of the third last block and so on. The first block is not encrypted, it is considered as the basic block. A *program controller* is built to contain the decrypting routine and is placed at the end of the program. Each block contains *pointer* to the *program controller* to decrypt the next block with its *hash value*. When the program executes, the basic block calculates its hash value and passes it to the program controller to decrypt the next block and continues its execution. The next block also does the same thing and this mechanism continues till the end. The hash values are calculated dynamically during program

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 249–253.

execution, so the adversary cannot alters the program statically as the binary codes are in encrypted form after protection [1].

III. Hardware based solutions

The most common hardware based protection approach uses a trusted processor, a tamper-resistant hardware, which checks and verifies every piece of software that is requested to be run on the computer [4, 6]. The hardware, for example, stores all of the keys necessary to verify digital signatures, decrypt license files, decrypt software before running it, and encrypt messages during any message transfer. The same software or media could be encrypted in a different way for each trusted processor that would execute it, because each processor would have a distinctive decryption key. This would put quite a dent in the piracy problem, as disseminating your software or media files to others would require the same hardware, which the software pirates cannot produce.

The other approach is to associate the software with the hardware of a particular machine. So that when the software runs, it secretly sends the serial number associated with the hardware to the software company [2]. In yet another approach, a movable piece of hardware, such as a *smart card* or a *dongle* or a *secure digital memory card*, is used to protect the software. The software is prevented from running unless some specific information from the trusted hardware is retrieved [3, 5, 8].

In addition, all the hardware based protection mechanisms suffer from the general drawback of lacking user friendliness, which reduce the importance of using such mechanisms for software protection.

IV. Checksums

A straightforward technique is to calculate the checksum of a program using any message digest algorithm and integrate it with the program. When the program is run, a checksum of the original program is calculated and compared with the already computed checksum. If they both match, the program is considered unmodified [8]. However, once the checksum is detected in a program, an attacker can remove them or patch around them.

The mechanism of integrating checksum within a program is also used in software tamper resistance checks of applications through dynamic monitoring. Two different processes are created, *Monitor* or *M – process*, and *Program* or *P – process*. The *M – process* is designed explicitly to monitor the control flow of the main program process. The *P-process* sends information on its instantiated control flow at a fixed period to the *M –process*. If there is a violation of the control flow conditions captured within the *M – process*, the *M–process* takes an anti-tamper action such as termination of the *P-process*. The length and checksum of the *P–process* is integrated in the *M–process*. When the program is run, the *M–process* computes the length and checksum of the *P–process* to ensure that an adversary has not added some extra instructions to the *P–process* [7].

V. Obfuscation

Obfuscation transforms a program into another program that has an equivalent behavior but which is harder to understand [9, 10, 11]. Obfuscation has been proposed as the solution to problems such as protection of transient secrets in programs, protection of algorithms, license management for software, and protection of digital watermarks in programs, software-based tamper resistance, and protection of mobile agents [12].

Obfuscation transformation quality is a combination of four measures; *potency*, which measures how much obfuscation is applied to the program; *resilience*, which measures how well the transformation holds up under attack from reverse engineering tools; *stealth*, which measures how well the obfuscated code blends in with the rest of the program; and *cost*, which measures performance penalty which a transformation incurs on an obfuscated application [19].

A research was conducted on whether automated obfuscation tools can produce obfuscated code that is resistant enough to analysis so that deobfuscation always requires significant manual analysis (or manual guidance of deobfuscation tools). If those techniques are sufficient to force a manual component to deobfuscation, they have provided a positive cost/benefit tradeoff between obfuscation and deobfuscation since the obfuscation techniques were automatically applied without human involvement, and are therefore relatively inexpensive [12].

An experiment was conducted on varying degrees of obfuscated programs and it was found that reverse engineering tools performed better on un-obfuscated codes as compared to the obfuscated ones [32]. An approach of using class *De-compilers* to provide security of Java applications, used in software *REVEAL* [14], is to first construct source code through De-compiler and then apply some obfuscation transformations to the generated source code. A theoretical basis for software obfuscation was provided on the basis of complexity problem, where proposed techniques were implemented using a prototype tool [16]. It was proved that a general obfuscator cannot be created that can transform a vast number of applications and perform a variety of tasks; instead, specific obfuscators should be created for performing specific tasks [15].

In order to hide secrets in software implementation, several software obfuscation techniques have been proposed [17, 18, 6, 13, 19]. Obfuscation may have an impact on performance; to balance that software inspection [22] can be performed to enhance the quality of software before obfuscation.

VI. GUARDS

A guard is regarded as a small code segment that performs checksums on part of the binary code to detect if the software has been modified. The guards are inserted into the software at different locations. They are inter-related so as to form a network of guards that reinforce the protection of one another by creating mutual-protection [23].

A programming logic for Java byte code is defined; where guards are used as a basis to allow proof of byte code programs containing loops. A guard is associated with each loop that is evaluated before loop's execution. Guard's value is associated with programming logic, which changes as the program is modified. If any modification is done in the program, the program counter's value will change that will ultimately be detected by the guard [24].

Guards can also be used in guarded inlining technique that uses runtime tests to ensure the correctness of optimized program. The optimized code preserves the semantics of original program regardless of the input data and the execution environments. However, code patching, pre-existence based inlining, and on-stack replacement present a new scenario for optimizations, where optimized code is correct only with respect to the execution environment at optimization time. These techniques ensure that the system has ability to correct the invalidated code if future execution violates optimization assumptions. However, if the assumption is most likely to be true in the future, speculation enables more effective and aggressive optimizations [25].

VII. SOFTWARE AGING

It is a technique in which a periodic update of the software, compatible with the older versions, is sent to the customer [6, 26]. Updates offer bug fixes and new features to users as well as keep program in synchronization with other programs on which the software depends. However, this approach is useful only for document centric applications, such as Microsoft Word, that rely on data being in specific format. The software aging modifies format from one version to the next. Cryptographic techniques help ensure that the later version of the software can use the older versions of documents, but not vice versa [8].

VIII. CRYPTOGRAPHIC TECHNIQUES

Several attempts have been made to prevent de-compilation of Java class files by encrypting class files so that nothing can read them except for the target Java Virtual Machine (*JVM*). The developer first encrypts the class files to secure them. When the application is executed, the encrypted class files are loaded by a custom *ClassLoader,* which decrypts the class files just before passing them to the *JVM*. This approach has a number of holes. At the very least, a compromised *JVM* can simply output the decrypted class files for later analysis. But there are also several places where the encrypted file is no longer encrypted and is vulnerable to attack. For example, the custom *ClassLoader* program can be decompiled, modified so that the decrypted file can be captured as a stream of byte code, and recompiled so that it dumps the class file just before it is passed to the JVM. Other problems are related to key security because the cryptographic key needs to be part of the application so that you can decrypt the classes in the custom class loader. If the hacker can find the key, then they can decrypt your class files before they get into the class loader [9].

An approach of embedding cryptographic information, such as *private key, digital signature algorithm*, etc., in hardware is used to digitally sign a Java class file. A suitably modified *JVM* can use this signed information and bypass verification and speed up linking process [28].

Likewise, cryptography offers very little help to mobile code. By using *Digital Signatures*, cryptography can be used to certify the origin of mobile code, and provides guarantee that the code received was not tampered while in transit. Cryptography can make no guarantees about what the code might do when executed [29].

IX. WATERMARKING

It is not always possible to prevent attackers from illegally reverse engineering the software. A mechanism is, therefore, necessary to let authors prove their ownership of the software. The goal of watermarking is to embed secret message into the software in a manner that makes it hard to remove by an adversary without damaging the software's functionality [4]. *Software fingerprinting* is a similar technique that embeds unique *customer identification number* into each distributed copy of software in order to facilitate tracking and prosecution of copyright violators [3]. Software watermarks can be static, can be produced without running the application; or dynamic, requires program to execute to retrieve watermarks.

Several watermark techniques have been proposed and implemented to prove ownership of software [2, 13, 30, 31].

X. Comparison of the Techniques

In this paper, some software protection techniques have been discussed. These techniques can be used in different situations; all of them may not be used together. The multi-block hashing scheme is uneconomical in real time situations. Though, it proves to be a quite good technique in a sense that a reverse engineer cannot decompile it statically. The hardware based protection mechanisms are expensive and suffer from the general drawback of lacking user friendliness, which reduce the importance of using such mechanisms for software protection. But, they also protect the software sincerely as there is a specialized hardware for each of the software that is distributed. The checksums does not seem to be an ample choice as a reverse engineer can easily patch around them. Code obfuscation seems to be a promising choice as it changes the look of the code by making it appear more complex, without changing the behavior. Guards also serve a good purpose in protecting the software. As these works with the checksums and successful program execution depends on their calculation, these can be patched around. Software aging is a good choice only for data centric applications; they do not serve any purpose in other types of applications. Cryptographic techniques do not prove to be a good choice as the software vendor has to provide the decrypting routine and the key in order to make the software run. Once the decrypting routine and key is distributed, the software can easily be reverse engineered. Software watermarking is used in situations where it is not always possible to prevent reverse engineering attacks. They are used to prove ownership of the software. These are poured in the software in a manner that it becomes difficult to identify them. Thus, they lead to prove ownership of the software at a later time.

XI. Conclusions

In this paper, we reviewed some of the major hardware and software based techniques intended to protect the software. The aim of these techniques is to detect attempts to tamper with software, and to protect the software against such attempts. Although there is no guarantee that the software is completely safe against reverse engineering, misuse, piracy and tampering; but the objective is to make the process of attacking the software difficult and time consuming.

References

[1] P. WANG, *Tamper resistance for software protection.* M.S. thesis, School of Engineering Information and Communications University, Daejeon, 2005.

[2] J. PALSBERG, S. KRISHNASWAMY, M. KWON, D. MA, Q. SHAO and Y. ZHANG, Experience with software watermarking. *Proceedings of the Sixteenth Annual Computer Security Applications Conference*, pp. 1-9, 2000.

[3] C. COLLBERG and C. THOMBORSON, Watermarking, tamper-roofing, and obfuscation – tools for software protection. *Proceedings of the IEEE Transactions on Software Engineering*, Vol. 28, No. 8, pp. 735-746, 2002.

[4] M.J. ATALLAH, E.D. BRYANT, and M.R. STYTZ, A survey of anti-tamper technologies. *The Journal of Defense Software Engineering*, pp. 12-16, 2004.

[5] P.C.V. OORSCHOT, Revisiting software protection. *Proceedings of the Sixth International Conference on Information Security*, Bristol, UK, pp. 1-13, 2003.

[6] L. ERTAUL, and S. VENKATESH, JHide – a tool kit for code obfuscation. *Proceedings of the Eighth IASTED International Conference Software Engineering and Applications*, Cambridge, USA, pp. 133-138, 2004.

[7] B. BLIETZ, *Software tamper resistance through dynamic monitoring*. M.S. thesis, Iowa State University, Ames, Iowa, 2004.

[8] G. NAUMOVICH and N. MEMON, Preventing piracy, reverse engineering, and tampering. *IEEE Computer Society*, Vol. 36, No. 7, pp. 64-71, 2003.

[9] G. NOLAN *Decompiling Java*, Apress, Berkeley, 280 pp., 2004.

[10] M.R. STYTZ and J.A. WHITTAKER, Software protection: security's last stand. *IEEE Security and Privacy*, Vol. 1, No. 1, pp. 95-98, 2003.

[11] D.E. BAKKEN, R. PARAMESWARAN, D.M. BLOUGH, A.A. FRANZ and T.J. PALMER, Data obfuscation: anonymity and desensitization of usable data sets. *IEEE Security and Privacy*, Vol. 2, No. 6, pp. 34-41, 2004.

[12] L. D'ANNA, B. MATT, A. REISSE, T.V. VLECK, S. SCHWAB and P. LEBLANC, *Self-protecting mobile agents obfuscation report*. Technical Report No. 03-015, Network Associates Laboratories, 2003.

[13] C. COLLBERG, G. MYLES and A. HUNTWORK, Sandmark - a tool for software protection research. *IEEE Security and Privacy*, Vol.1, No. 4, pp. 40-49, 2003.

[14] V. TAM and R.K. GUPTA, Using class decompilers to facilitate the security of java applications. *Proceedings of the 1st International Conference on Web Information Systems Engineering*, Vol. 1, 2000.

[15] B. BARAK, O. GOLDREICH, R. IMPAGLIAZZO, S. RUDICH, A. SAHAI, S. VADHAN and K. YANG, *On the (im)possibility of obfuscating programs*. Technical Report No. 57, Electronic Colloquium on Computational Complexity, 2001.

[16] T. OGISO, Y. SAKABE, M. SOSHI and A. MIYAJI, Software obfuscation on theoretical basis and its implementation. *Proceedings of the IEICE Transaction Fundamentals*, Vol. E86-A, No. 1, 2003.

[17] J. CHAN and W. YANG, Advanced obfuscation techniques for Java bytecode. *Journal of Systems and Software*, Vol. 71, pp. 1-10, 2004.

[18] C. COLLBERG, C. THOMBORSON, and D. LOW, *A taxonomy of obfuscating transformations*. Technical Report No. 148, Department of Computer Science, University of Auckland, New Zealand, 1997.

[19] C. COLLBERG, C. THOMBORSON and D. LOW, Breaking abstractions and unstructuring data structures. *Proceedings of the 1998 International Conference on Computer Languages*, pp. 28, 1998.

[20] C. LINN and S. DEBRAY, Obfuscation of executable code to improve resistance to static disassembly. *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 290-299, 2003.

[21] M. SOSONKIN, G. NAUMOVICH and N. MEMON, Obfuscation of design intent in object oriented applications. *Proceedings of the 2003 ACM Workshop on Digital Rights Management*, pp. 142-153, 2003.

[22] E. EMDEN and L. MOONEN, Java quality assurance by detecting code smells. *Proceedings of the 9th Working Conference on Reverse Engineering*, Amsterdam, Netherlands, pp. 97-106, 2002.

[23] H. CHANG and M. ATTALAH, Protecting software code by guards. *Proceedings of the ACM Workshop on Security and Privacy in Digital Rights Management*, Philadelphia, pp. 160-175, 2001.

This page intentionally blank

[24] C.L. QUIGLEY, *A programming logic for Java bytecode programs*. Ph.D.thesis, University of Glasgow, Glasgow, Scotland, 2004.

[25] F. QIAN, *Runtime techniques and interprocedural analysis in Java virtual machines*. Ph.D. thesis, School of Computer Science, McGill University, Montreal, 2005.

[26] M. JAKOBSSON and M.K. REITER, Discouraging software piracy using software aging. *Proceedings of the ACM digital Rights Management*, pp. 1-12, 2002.

[27] L. ERTAUL and S. VENKATESH, Novel obfuscation algorithms for software security. *Proceedings of the 2005 International Conference on Software Engineering Research and Practice*, Las Vegas, 2005.

[28] P.T. DEVANBU, P.W. FONG and S.G. STUBBLEBINE, Techniques for trusted software engineering. *Proceedings of the 20th International Conference on Software Engineering*, pp. 126-135, 1998.

[29] D.S. WALLACH, *A new approach to mobile code security*. Ph.D. dissertation, Princeton University, Princeton, New Jersey, 1999.

[30] G. ARBOIT, A method for watermarking Java programs via opaque predicates. *Proceedings of the 5th International Conference on Electronic Commerce Research*, Los Alamitos, pp. 124-131, 2002.

[31] A. MONDEN, H. LIDA, K. MATSUMOTO, K. INOUE and K. TORII, A practical method for watermarking Java programs. *Proceedings of the 24th Computer Software Applications Conference*, Taipei, Taiwan, pp. 191-197, 2000.

[32] L. VINCIGUERRA, L. WILLS, N. KEJRIWAL, P. MARTINO and R. VINCIGUERRA, An experimentation framework for evaluating disassembly and decompilation tools for C++ and Java. *Proceedings of the 10th Working Conference on Reverse Engineering*, 2003.

[33] P.T. Devanbu and S.G Stubblebine, Software engineering for security: a roadmap. *ICSE - Future of SE Track,* pp. 227–239, 2000.

[34] G. Hachez, L.D. Hollander, M. Jalali, J.J. Quisquater and C. Vasserot, Towards a practical secure framework for mobile code commerce. *Proceedings of the Third International Information Security Workshop, Wollongong,* Australia, *Vol. 1975 of LNCS*. pp.164–178, 2000.

[35] C. Wang, J. Davidson, J. Hill and J. Knight, Protection of software-based survivability mechanisms. *The International Conference on Dependable Systems and Networks,* Goteborg, Sweden*, IEEE Press*, pp. 193–205, 2001.

Code Characterization for Automatic User Interface Generation

Jaroslav Kadlec
Department of Computer Graphics and Multimedia
Faculty of Information Technology, Brno University of Technology
Bozetechova 2
612 66 Brno, Czech Republic
kadlecj@fit.vutbr.cz

***Abstract*—This paper presents extended taxonomy that can be used for better automatic user interface generation. Taxonomy is focused on code characterization allowing an artificial intelligence user interface generator to create user interface which is not limited by visual presentation of characterized data. This allows user interface to contain lots of various operation over characterized data creating rich and interactive user interfaces that could not be created with data characterization only.**

I. Introduction

Creation of user interfaces in various applications is getting more and more complicated. Users request high quality user interfaces and complex applications that are user friendly. Users also expect to have same applications on various devices such as phones, PDAs, notebooks and others. Creation of interface of application that can be portable between various platforms is very difficult, leading in creation of multiple user interfaces which are based on expected device's capabilities and features. Creation of such user interfaces is problematic, leading to increased time of application development. That is why a concept of automatic generation of user interfaces was developed.

Automatic user interface generation systems promise to simplify an application programmer's design tasks by providing a set of design rules [1] and effectiveness criteria [2]. To establish these criteria, it is necessary to understand which of the properties of the information to be visualized are related to user interface design and how they are related. This task is called data characterization [3, 4]. With flexible data characterization it is possible to create automatic presentation systems. These however do not allow creation of rich user interfaces. To create rich user interface with possibilities of various operations over the characterized data, a code characterization is required.

With complete data and code characterization, application programmer is able to describe application code that will be used for automatic user interface generation [5]. Because automatic user interface generation is very complex process requiring artificial intelligence, current user interface management systems are using designer made user interfaces [6, 7, 8]. Such user interface management systems benefit from current code which is independent on user interface which can be simply modified for use with various devices having different presenting capabilities. Also code can be modified independently from user interface design or its components which increases maintainability of application.

II. Previous Work

First data characterization taxonomy was proposed by Mackinlay [2] who was using data properties to guide automatic design of visual presentations. The taxonomy was primarily designed for quantitative data. This taxonomy was later extended by Roth and Matis [3] to address more complex quantitative data. Arens, Hovy and Vossers [9] developed a vocabulary that was able to describe multimedia information. Wahrend and Levis [10] introduced partitioning of data into several categories such as shape and structure. Zhou and Feiner [4] restructured taxonomy into six domains, introducing data role and data sense. While data role characterized data based on user information seeking goals, data sense represented user interpretation preferences.

III. Code Characterization

Code characteristic is expressed using annotation tags that are divided into five dimensions: *command category, command attribute*, *command parameter*, *command sense* and *event*. *Command category* represents set of operations or commands that can be executed with the object. *Command attribute* defines basic properties of the operations and defines various usage options. *Command parameter* describes properties and options for command parameters. *Command sense* distinguishes how should be commands treated. *Events* define optional reactions of user interface on internal object changes.

Code characterization is proposed for object-oriented environment and that is why each piece of information is supposed to be object. Each object belongs to data domain and has data type as defined for data characterization in [4] which

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 255–259.

is considered to be very complex for data characterization task and which was slightly modified and extended to support code characterization presented here. Each data characterized object can define public methods that can be characterized using code characterization. Because of object-oriented environment, every inherited object inherits public data and methods including data and code characterization. Because taxonomy for data and code characterization is flexible, new types of methods or relations can be easily added to the taxonomy. Following subsections describe code characterization taxonomy in more detail.

A. Command Category

Command category defines set of operations or commands that can be executed on the object. A typical example of command category can be set of methods that allow playing, stopping, pausing of recorded data (might be audio or video playback, or any other data that object can replay or record). Set of such commands have usually default symbolic representation and users are used to this concept from real world or other applications. Command categories can be easily defined in XML format and extend existing set of categories. Categories can be represented in user interface using concept of Smart-Templates [11] which contain information about default symbolic representation and also default graphical design. Because object can be of various categories, multiple categories can be defined for single object. Following categories are basic for most applications and should always be implemented.

1) Collection

Collection contains methods for adding, removing and selecting items contained in object. With this category it is possible to create various objects that act like collections of various data with possible multi-selections and other functions. Add or Remove methods can add or remove objects of various types and do necessary checking before added or removed object is processed. This functionality was almost impossible with data characterization only.

2) Storage

Storage defines methods for saving, opening, and creation of contents. Methods are usually represented by common symbols (e.g. save by diskette icon) and are very important in applications that want to add functionality of creation of new objects and their saving or loading.

3) Navigation

Methods in navigation are most usually used for moving of cursor in collections. Navigation defines actions for previous or next item, first or last item. Navigation can be defined separately from collection, because it can control cursors in various collections at once. A typical example is media player with playlists: user can select one playlist and let play next media file from selected playlist.

4) Media Player

This category defines methods for start (play), stop and pause of playback. This methods are usually represented by standard symbols and should be always implemented in default order.

5) Clipboard

Clipboard is widely used technique for data copying and moving. Clipboard category describes methods that can be used for this purpose and that is why user interface can offer this functionality to user. Typical clipboard methods are copy, paste and cut, storing selected data to clipboard for other applications that may use them.

6) Draggable

Object with method that belongs to draggable category informs that object can be dragged. When object can be dragged, user interface allows user to drag certain objects to another objects. Altough object can be data characterized to be draggable, it is missing method that could process the drag request. This method now can be characterized using code characterization.

7) Droppable

Object containing methods with droppable category contain logic of checking whether dragged object can be dropped and acquiring dropped object. Because drag and drop operations require some logic that cannot be achieved by data characterization, draggable and droppable categories are needed.

B. Command Attribute

Attributes express various information about methods that are implemented by the objects. Although it might seem that information according to each of the method are very different, they have quite lots of common attributes that need to be defined for proper creation of user interface. If some object implements a method that should be visible in user interface, basic attributes such as *name*, *description*, or *importance* should be defined.

1) Name

Name attribute contains name of the method that should be presented to user in user interface. Name is not required for methods that already have its category, because it already has name attribute. Name attribute can also be specified in multiple languages so that user can choose which of the languages he prefers.

2) Description

Description describes method and acts like a tip for the user. When user wants to call certain command from menu or from other part of user interface, system can display or play short help for the command based on the capabilities of the device or user preferences.

3) Importance

Some commands are more important than others. It is good practice to make such commands more accessible. In graphical user interfaces, important commands are placed to toolbars or tool strips so that user can easily access them. Importance attribute represents how important some commands are and whether it should be somehow highlighted or placed in user interface to a location where it is easily accessible.

4) Representation

Representation defines a symbol that represents certain commands. Representation is especially important for commands that have high importance. Representation can be defined as icon or image. For commands that belong to a category is representation most usually defined in category definition file. In this case, another definition of representation overrides default representation.

5) Dependence

Every method in the code has some dependencies that must be satisfied before the code is executed. This is most usually checked by developer programming user interface. In automatic user interface generation, user interface has to know when is command enabled or disabled. So dependence expresses conditions which have to be satisfied to allow execution of selected command. Dependence is very important because without proper dependence checking, user can have access to unavailable commands. An example can be playing of media file although there is no media file open.

C. Command Parameter

Almost every implemented method has some parameters. Parameters init method and method most usually works with available parameters or internal variables. That is why its characterization is very important. Following parameters are basic parameters that are important for proper calling of selected method. An example can be drawing to image. Method that can draw a line has three main parameters: color, starting point and ending point. User interface based on parameter characterization knows that form, which is displaying the image, has a color palette to which is related color in draw line method. User interface will not require user to specify this parameter manually and take the value from the color palette. Because points are data characterized and related to image, user interface knows that the location should be taken from the displayed image and when user selects draw line command, user interface expects user to select two points in the area. After both points are selected, command is executed. Following parameters are important for proper parameter characterization.

1) Name

Describes name of the parameter and is similar to name in command attribute. Name is not always required but is very important in user interfaces that are not based on visual interfaces (e.g. for blind people).

2) Description

Description acts as a tip for the parameter. It is also similar to description in command attribute.

3) Relation

Relation is important for the parameters because it describes relation between parameter and another object in environment. When relation is specified, user interface automatically takes result of relation as input for the parameter. An example is color from color palette. Drawing methods require color which is taken from color palette automatically.

4) Default value

Some parameters, most usually quantitative, require certain value. Although user can change the value, sometimes it is more convenient to show user some kind of default value that can be used for the command too. An example is number of passes of some computational task. Although user can specify much greater value, algorithm can have good results in five passes. Maximum and minimum values are boundaries which can be specified in data characterization for the parameter data type [4].

D. Command Sense

Sense helps to distinguish how are methods executed. By default, user selects command, user interface asks for parameters and method is executed with all specified parameters. Such default sense is called *command*. When *tool* sense is specified, user interface runs method again and again as long as the command in context is selected. Typical example is selecting some tool – e.g. a draw line tool – from toolbox. As long as user keeps up with specifying points in the image or as long as the tool is selected, user interface calls draw line command repeatedly and user draws lines. *Command* and *tool* sense cover most of the types of method executions and that is why various types of user interfaces can be generated.

For some objects there also exists default action that can be executed whenever user selects the object and launches default action. In graphical user interfaces this is most usually done by double-clicking. For blind users, there is usually voice command *run* or *open*. In this case, it is possible to specify which command is default and should be used in such cases.

E. Event

Events represent a mechanism that informs user interface or application code that there have been some changes in the data from the user or from the application code. Although various approaches exist for implementing such behavior using callbacks [12], TAPS [1] or ORB [8], event mechanism is very simple and works well for duplex communication. Base event, that should be implemented for every object is *changed*. Such event should be executed always when application code changes data of certain object. Because user interface cannot detect changes in data without having checked every available public property, signaling data change using change event is significant to increase performance of application. For data, that have data characterization with high dynamic transience [4], user interface can check its state very often and optimize the performance. Event mechanism can be used for static data or data with low dynamic transience.

Because user interface set the data that are changed in user interface immediately to the object's properties, it is not necessary to inform object about changes in public data. However for languages that do not support concept of properties (e.g. C++) it is necessary to implement set method that should be characterized by *userChanged*. This event contains also definition for what data is event designed so that user interface can react correctly when user changes any data.

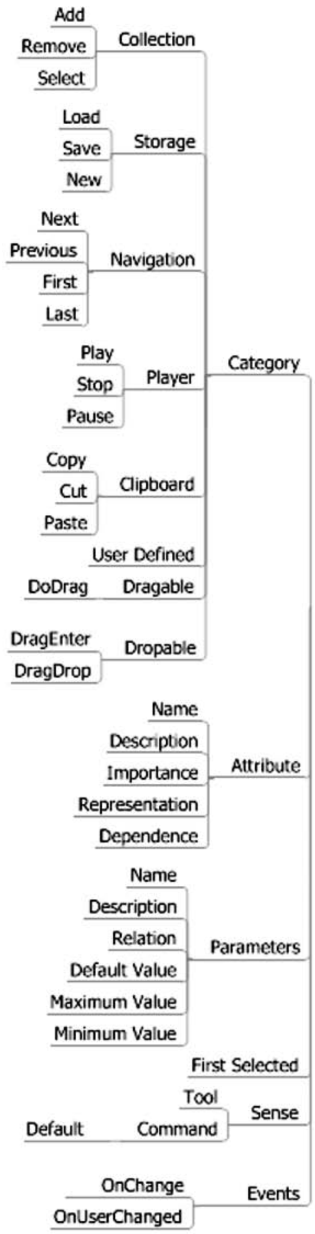


Fig. 1. Code characterization taxonomy.

IV. EXAMPLE

As an example imagine a media player. Media player consists of *playlist* with media files to play, *play*, *pause*, *stop*, *next*, and *previous* buttons, and a *timeline* to see and jump into selected part of media file. Media player also has functions allowing *adding* of media files and directories and *removal* of selected items from the playlist or clearing the list.

Method *AddMediaFile(string mediaFile)* can be code characterized by *Category(Collection(Add, playlist))* so that method is add method for the playlist and user interface can group this list with this command together. *Attribute(Name("Add media file"))* describes name to present to the user, *Attribute(Description("Adds selected media file to playlist."))* is a tooltip for the user, *Attribute(Importance(High))* to have the control quickly available. Representation remains the same as specified in category and there is no dependence for the addition of media file. *Parameter(mediaFile, Name("Media File"))* defines name of parameter, data characterization consist at least of *type(Atomic)* because path to file is not divisible, *domain(Entity(Path("*.mp3"))* defines that string is a path that should be specified by the user. With this data and code characterization, user interface will show user a default exploring window and allows selection of files with .mp3 extension. *AddMediaFile* method loads files that were selected by the user from explorer window and launches event *OnPlaylistChanged* which is characterized as *Event(OnChanged(playlist))*. This will let user interface update contents of the list. Method that would allow addition of whole directory can be characterized alike except that mask for the Path will be different.

Play method that starts playing of selected item in the list will be of *Category(MediaPlayer(Play))* so there is no need for name, description or representation specification as it is already done in category. Play is dependet on method that checks if list contains items - *Attribute(Dependency(IsSelectedItem))*. When object changes, this method must be reevaluated to enable or disable Play command.

Now when user adds items to the list using Add media file command, Play button and other buttons that are dependent on *IsSelectedItem* will be available and user can start playing of media files in the list.

Although code and data characterization for media player above is not complete, it is now possible to have simple media player with custom media file loading and playing.

V. CONCLUSION

This paper presents basic taxonomy that can be used for code characterization – a complement to data characterization. Code, following design rules for data and code characterization, benefits from user interface independence, easy maintainability and extendibility. User interface can be generated by artificial intelligence user interface generator or made by human user interface designer. Such concept leads into possibilities to have user interfaces apart from application code. Each user interface can be designed for certain device it will be used on. Same code can be easily transferred between phone, PDAs, computers and virtually any future devices without problems.

This page intentionally blank

Code characterization offers way how to implement rich user interfaces. In contrast, user interfaces created from complex data characterization allow only direct manipulation with characterized data, but it is impossible to create various operations with characterized data. Code characterization process allows creation of actions and operations over characterized data and creation of fully interactive user interfaces.

Now, main goal in code characterization is to create high quality artificial intelligence that would be able, such as human user interface designer, use code and data characterization to create fully functional user interface from one application data on multiple devices. Artificial intelligence will follow same design rules that have been tested during creation of example applications by human user interface designer.

VI. Acknowledgement

This work has been supported by research grant Rapid prototyping tools for development of HW-accelerated embedded image- and video-processing applications from Grant Agency of the Academy of Sciences of the Czech Republic, No. 1ET400750408.

References

[1] Berlege, T., Using TAPS to separate the user interface from the application code, ACM SIGGRAPH UIST'92, pages: 191-198, 1992
[2] Mackinlay, J., Automating the design of graphical presentations of relational information, ACM Transactions on Graphics, pages 110 – 141, 1986
[3] Roth, S. F., Mattis, J., Data characterization for intelligent graphics presentation, In Proceedings of the 1990 Conference on Human Factors in Computing Systems, pages 193–200, New Or-leans, LA, 1990. ACM/SIGCHI
[4] Zhou, M. X., Feiner, S. K., Data Characterization For Automatically Visualizing Heterogeneous Information, In Proceedings INFOVIS '96(1996 IEEE Symposium on Information Visualization) page 13-20, San Francisco, CA, October 28–29, 1996
[5] Jelinek, J., Slavik, P., GUI Generation from Annotated Source Code, In Proceedings of Tamodia'04, Prague, Czech Republic, 2004
[6] Bernstein, P. A., *Middleware, an architecture for distributed system services*, Cambridge Research Labs, CRL 9316
[7] Jokstad, H., Sundling, C. V., *Picasso-3 User Interface Management System*, Presented at the Enlarged Halden Programme Group Meeting, Storefjell, Norway, 8-13 September 2002, HPR-358
[8] Nihe, K., Seki, K., Nakamura, H., Yagishita, R., Shimojima, T., *Distributed object system framework ORB*, NEC Research and Development, pages 292-297, 1994
[9] Y. Arens, E. Hovy, and M. Vossers. The knowledge underlying multimedia presentations. In M. Maybury, editor, Intelligent Multimedia Interfaces, pages 280–306. The MIT Press, 1993
[10] R. Wehrend and C. Lewis. A problem-oriented classification of Visualization techniques. In Proceedings of the First IEEE Conference on Visualization: Visualization 90, pages 139–143. IEEE, Los Alamitos, CA, October 1990
[11] J. Nichols, B.A. Myers, K. Litwack, Improving Automatic User Interface Generation with Smart Templates. In Intelligent User Interfaces, Funchal, Portugal: pp. 286-288, 2004
[12] Pope, G. E., Krasner, S. T., A cookbook for using the Model-View-Controller user-interface paradigm in Smalltalk-80, Jour of Object-oriented programming 1 (3), pages 26-49, 1988

Lattice Cube Semantic Index Based Mining on XML Documents

Dr. A.M. Natarajan
Principal
Kongu Engineering College
Erode, Tamilnadu
India

principal@kongu.ac.in

K. Premalatha
Assistant. Professor / CSE-PG
Kongu Engineering College
Erode Tamilnadu
India

kpl_barath@yahoo.co.in

A. Kogilavani
Final M.E.
Kongu Engineering College
Erode, Tamilnadu
India

vani_sowbar@yahoo.co.in

Abstract - XML (eXtensible Markup Language) is fast becoming the de facto standard for information exchange over the Internet. As more and more sensitive information gets stored in the form of XML, sophisticated indexing schemes are required to speedup document storage and retrieval. XML documents can be hierarchically represented by elements. This paper describes a Lattice-map semantic indexing technique to cluster XML documents. To improve performance of information retrieval, documents can be indexed using Lattice-map technique. Similarity and Popularity operations are available in Lattice-map indexing technique and clustering algorithm is used for mining XML documents.

Index Terms : Lattice-map indexing, Lattice Cube, XML, Path-Document Matrix

1. INTRODUCTION

The increasing relevance of the Web as a mean for sharing information around the world has posed several new interesting issues to the computer science community. The traditional approaches to information handling are ineffective because they are mainly devoted to the management of highly structured information, like relational databases, whereas Web data are semi structured and encoded using different formats like HTML, XML, email messages and so on.

The eXtensible Markup Language (XML) is a W3C standard for presenting and exchanging information on the Internet. In recent years, more and more areas are adopting the XML standard to represent their information from XML documents, a number of XML query languages have been developed. Sophisticated indexing schemes have been proposed to speed up document storage and retrieval. However, because the size of XML documents is very large and the types vary typical information retrieval techniques such as LSI (Latent Semantic Index) are not satisfactory. Information retrieval on the Web is not satisfactory due to partly poor usage of structure and content information available in XML documents.

Many Web applications that process XML documents, such as grouping similar XML documents and searching for XML documents that match a sample XML document, will require techniques for clustering and classifying XML documents. It has been well-established in such fields as database management and information retrieval that the more semantics about data are understood by a system, the more precise queries can become. It is intuitively obvious that if some of the rich semantics of XML can be taken into account, it is more powerful basis of supporting the clustering and classification of XML documents for a wide variety of XML applications.

Consider a document databas (D). Each document (d) is represented in XML and it contains XML elements. Each element has zero or more terms bound to it. Typical indexing requires a frequency table that is a two-dimensional matrix indicating the number of occurrence of the terms used in documents. By generalizing this idea, this paper introduces Lattice Cube that consists of triplet (d, p, c). Here p is an XML path and c is a concept. In such context, we address the problem of clustering structurally similar XML documents. This problem has several applications like recognizing different sources providing the same kind of information and in the structural analysis of a web site. Grouping semi structured documents according to their structural homogeneity can help in devising indexing techniques for such documents, thus improving the construction of query plans. This paper describes a new Lattice-map indexing based technique which is referred to as “Lattice Cube”, that represents the triplet (d, p, c) where d represents document, p is an XML path and is concept and clustering technique that can cluster such documents semantically. Before going further, consider the following example.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 261–266.

1.1 Motivating Example

```
(1) <section>
(2)  <section> XML is represented in a
               Lattice-map indexing …
(3)   <section> it is a new standard …
      </section>
(4)   <section> An application is as shown in
(5)    <figure>
          http://www.a.b.c/clustering.algs
       </figure>
(6)    <caption> Clustering Algorithm
        </caption>
(7)   </section>
(8)  </section>
(9)  <section> Lattice-map indexing
     technique … </section>
(10)<verticalskip/>
(11)</section>
```

Figure 1: XML Document

Suppose that a query Q1 is posed to find all documents that describe “Indexing” in more than one sub-subsection. Notice that this type of queries asks for a specific document structure that is not for section, nor for subsection but for sub-subsections. Searching an entire XML database is costly because a word pattern for search is rarely used if we search against a large document database. That is, a word list for a document is sparse as compared to the list of words available in the database. Search for a sparse list of words is not efficient. To resolve this problem, this paper proposes a way of clustering XML documents semantically. In this way, searching can be restricted within only a cluster, instead of all documents in order to improve the performance.

1.2 Organization

The remainder of this paper is as follows. Section 2 describes preliminaries such as element paths in XML documents, path-document matrix, popularity of a path column, finding radius and center. Section 3 introduces preprocessing steps, Lattice Cube that represents a set of triplets (document d, XML element p, terms or contents c). Section 4 describes XML document clustering based on Lattice-map indexing.

2. PRELIMINARIES

This section defines the following technical terms.

Definition 1: ***(Element Content)*** An XML element contains (1) simple content, (2) element content, (3) empty content, and (4) reference content.

As an example, consider an XML document as shown in Figure 1. The element <section> in line (9) has a simple content. The element <section> in line (1) has element content, meaning that it contains two subsections as shown in lines (2) and (9). Of course, two content types can be mixed, e.g., the element <section> in line (2) contains a simple content in line (2) and also elements in lines (3)-(8). The element <verticalskip> contains empty content. The content <figure> has reference content that hyperlinks to a site.

Definition 2: (*ePath*) Element Path, called “ePath,”[8] is a sequence of nested elements where the most nested element is simple content element. For example, in Figure 1 section.section.section.figure is an ePath, but section itself is not an ePath due to the top element <section> does not have simple content An XML document is defined as a sequence of ePaths with associated element contents. An XML document database contains a set of XML documents. This paper proposes a Lattice-map index for an XML document database. In a document ePath Lattice-map index, a path column represents an ePath, and a row represents an XML document.

```
d1:
<e0>
 <e1> V1 </e1>
 <e1> V11 </e1>
 <e2>
 <e3> V2 V3 V5 </e3>
 <e4> V3 V8 </e4>
 <e5 />
<e5>V12</e5>
</e2>
</e0>
```

```
d2:
<e0>
  <e1> V1 </e1>
  <e1> V11 </e1>
  <e2>
  <e3> V3 V7 </e3>
   <e3> V3 V8 </e3>
   <e4> V9
    <e5/>
  <e6> V4 </e6>
  <e7> V6 </e7>
  </e4>
  </e2>
   <e8> V6 V12 </e8>
  </e0>
```

```
d3:
<e0>
 <e1> V11 </e1>
 <e2>
 <e3> V2 V7 </e3>
 <e4>
  <e7>V3 V9</e7>
 </e4>
 <e5 />
<e5>V4</e5>
</e2>
<e9> V5 </e9>
</e0>
```

Figure 2: Example of XML Documents

Figure 2 is a set of simple XML documents. Define ePaths as follows: p0=e0.e1, p1=e0.e2.e3, p2=e0.e2.e4, p3=e0.e2.e5, p4=e0.e2.e4.e6, p5=e0.e2.e4.e7, p6=e0.e8, p7=e0.e9

2.1 Path-Document Matrix

Each ePath in a document is counted and it is placed in the path-document matrix. For each ePath, documents can be represented as shown in Figure 3.

	p0	p1	p2	p3	p4	p5	p6	p7
d1	2	1	1	2	2	0	0	0
d2	2	2	1	1	1	1	0	0
d3	1	1	1	2	0	1	0	1

Figure 3: Lattice-map index for Figure 2

Definition 3:(Hamming Distance) The distance between two documents can be defined as dist(d_i, d_j) =

$$\begin{cases} 1 & \text{if } p(d_i) = p(d_j) \\ \dfrac{|p(d_i) - p(d_j)|}{\mathrm{Max}(\, p(d_i)\, ,\, p(d_j)\,)} & \text{if } p(d_i) \neq p(d_j) \end{cases}$$

2.2 Indexing XML Documents

Each document d in an XML document collection X is represented in XML. Document d contains one or more concepts c that is associated with XML paths p.

Definition 4: (*Lattice-map Index of Documents: DI*). A collection of XML documents, X = {(d, p, c)}, is encoded into a Lattice-map index, D x (P x C), which is called "Document Lattice-map Index" or DI = (d, p, c, b) where d, p and c are an element of D, P, C respectively.

- D is a document collection, d Є D
- P is a set of XML path expressions, p Є P. Particularly, p is an XML path from the root element to a leaf element.
- W is a set of element contents or words, each of which is associated with p, w Є W
- b is the count of concepts associated with an XML path p in a document d.

2.3 Popularity of Path Column

A path column in a Lattice-map index can be described by its popularity. It is popular if used frequently enough. The index for the most popular path column is mode in a path-document index.

Definition 5: (***Popularity***) The popularity of a path column is pop(p) = $\sum p(d_i)$ / n where p (d_i) is path column. Calculate median of popularity path column values which is referred to as m. We can classify path columns into two cases. (1) If pop (p_i) > m then it is called "popular path column". (2) If pop (p_i) < m then it is called as "Unpopular path column".

2.4 Center

This section describes one of the features of Lattice-map indexes such as center

Definition 6: (***Center***) In a cluster of XML documents, the center is a vector where each element of the vector is the mean value of the corresponding path columns of the documents.

2.5 STANDARD DEVIATION AND CORRELATION COEFFICIENT

The standard deviation (*S*) is a measure of the difference from the mean [7]. Large value for *S* means the data is spread widely around the mean. Units are the same as the data itself.

$$S_i = \frac{1}{n}\sqrt{\Sigma \mathrm{dist}(d_i,\mathrm{center}(d_i))^2}$$

The statistical definition of "relatedness" of two documents, d_i and d_j, is called correlation. We can calculate a "correlation coefficient" F as follows:

$$F(d_i,d_j) = \Sigma\ \mathrm{dist}(d_i, \mathrm{center}(d_i)) \times \mathrm{dist}(d_j, \mathrm{center}(d_j)) \times \mathrm{MAX}([d_i],[d_j]) \times S_i \times S_j$$

3. Three Dimensional Indexing

3.1 Document Pre-Processing Steps

Pre-processing is a process in which the meaningful terms are extracted. Pre-processing involves the reformatting, or filtering, of a text document, to facilitate meaningful statistical analysis. In doing the Preprocessing on a text document the system will follow the following steps:

- Numbers and punctuation marks standing alone should be cut; for eg,"55", "720", "90", "4", " - "
- It is necessary to cut punctuation marks at the end of the words; e.g., the program should not distinguish the words "school" and "school," "colleges" and "colleges," from each other.
- The Program should not be sensitive to the case and all the words should be examined as low-case words; e.g., if the text contains words "school" and "School", the frequency of the word "school" in the output of the program should be two.
- It is important to exclude "stop" words - articles, preposition, unions etc. - which cannot be used as descriptors of any document (words like "and", "the", "when", "though", "for" etc.). To achieve that goal, a "stop list" of such "noise" words in additional file is prepared.
- One more aspect of pre-processing is correct counting of the words in specific grammatical forms. E.g. "broke" - "break", "broken" - "break", "phenomena" - "phenomenon". By using this procedure for the text containing words "break" and "broken" the program gives a frequency of two for the word "break.

There are lists of inflectional endings, based on syntactic category that can be detached from individual words in an attempt to find a form of the word that is in WordNet. There are also exception list files, one for each syntactic category, in which a search for an inflected form is done. Morphy tries to use these two processes in an intelligent manner to translate the string passed to the base form.

3.2 Concept Extraction

The people and information systems may use different conceptualizations of the "same" phenomena, use different terms for the same concepts(synonyms), use the same terms for different concepts(homonyms), or they use different units, measures and formats, or apply different deduction or reasoning rules. Common relations include

- Hypernym (Is a) – the taxonomic relation.
- Meronym (Part of) - the component relation.
- Synonym - the identity relation.

The concepts are extracted from the pre processed document by applying the algorithm.

- Tag a document;
- Select chaining candidates, $C_w = \{ W1, W2, \ldots, W_n \}$, and load them into the clustering candidate array processing identity relation;
- Initialize the current candidate pointer i;

```
while(i<n)
        for (j = i + 1; j < n; j++) do
                If HasRelation(Wi,Wj) defined in
                WordNet where Wj ε Cw
                        Link(Wi,Wj);
        end for
increase the current candidate pointer i
end while
```

The present work uses WordNet, which has fewer cross-category connections. The four kinds of relations are identity, synonymy, hyponymy (hyponymy), and meronymy. Hypernymy and hyponymy are regarded as one relation because they are inter-definable. Clustering is achieved by the above four inter-noun relations. To consider more relations it is necessary to manage more parameters when calculating the word weights, which would degrade the information retrieval performance. Therefore, to reduce the range of the problem, we used only a restricted set of relations in the present work to show that the proposed approach enhances IR performance. Clustering information is obtained from WordNet. Because the verb file has no relation with the three other files.

3.3 Lattice Cube

XML document is defined as a set of (p, c) pairs, where (1) p denotes an element path (or ePath) described from the root element, (2) c denotes a concept for an ePath.

Typical methods of handling text-based documents use a frequency table or inverted (or signature) file that represents words for documents. However, since XML documents are represented by XML elements (or XML tags), the typical methods are not sufficient.

In this section a 3-dimensional lattice-map representation, called Lattice Cube is introduced. A Lattice Cube for XML documents is defined as Lattice

Cube = (*d, p, c ,b*), where *d* denotes XML document, *p* denotes ePath, c denotes concept for ePath, and *b* denotes the count of the concepts associated with the path.

A Lattice Cube for a set of documents: {d1, d2, d3, d4, d5}.Each documents d1={(p0, c1), (p1, c2), (p1, c3), (p1, c5), (p2, c3), (p2, c8) }, .., d3={(p0,c11), (p1, c2), (p1, c7), (p2, c3), (p2, c9) …, (p_i,c_i2), (p_i,c_i3), (p_i,c_i4), …, (p_i,c_{ij})}, and so on. Path columns for ePath are organized in the same order as the order in which the documents are processed.

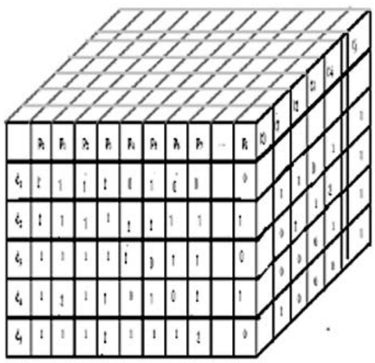


Figure 4: Lattice Cube (Partial Representation): Example

4. Document Clustering

A set of XML documents can be partitioned into *n* clusters. The number of partitions depends on the characteristics of documents; that is ePath and concepts used in the documents. In this section, for simplicity, only consider Lattice-map indexes representing documents and ePath. To cluster XML documents partitioning algorithm is used.

There are three approaches of identifying similarity:

- Clustering by computing similarity for pairs of documents at a time.
- Clustering by computing similarity of a Lattice-map with the center of a target Lattice-map index.
- Clustering by computing radius of a Lattice-map from the center of a target Lattice-map index. All Lattice-maps within a given radius are in the same partition.

4.1 Clustering Based on Center of Target Lattice-Map

This section describes the simplest unsupervised learning algorithm that solves the well known clustering problem. The procedure follows a simple and easy way to classify a given document collection through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given document collection and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same document collection points and the nearest new centroid. A loop has been generated.

As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.

2. Assign each object to the group that has the closest centroid.

3. When all objects have been assigned, recalculate the positions of the K centroids.

4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

5. Conclusion

The main contributions of this paper are (1) the application of Lattice-map indexing to represent XML document collection as a 3-Dimensional data structure: XML document, XML-element path, and concepts, (2) the definition of Lattice Cube index based schemes to partition documents into clusters,(3) the usage of unsupervised learning algorithm to cluster XML documents based on centroids.

6 REFERENCES

[1] E. Bertino, G. Guerrini, and M. Mesiti. Measuring the structural similarity among XML documents and DTDs. Technical report, Tech. Report DISI-TR-02-02, Department of Computer Science, University of Genova, 2002.

[2] Bo-Yeong Kang , Sang-Jo Lee Document indexing: a concept-based approach to term weight estimation

[3] C. Chan and Y. Ioannidis, Bitmap Index Design and Evaluation, Proc. Of Int'l ACM SIGMOD Conference, 1998

[4] A. Doucet and H. A. Myka. Naive clustering of a Large XML document collection. In Proc. 1st Annual Workshop of the Initiative for the Evaluation of XML retrieval Schloss Dagstuhl, Germany, 2002

[5] D. Guillaume and F.Murtagh. Clustering of XML documents. Computer Physics Communications, pp.215–227, 1989.

[6] P. Willet, Recent Trends in Hierarchical Document Clustering: a Critical Review, Information Processing and Management, 24:577-97, 1988

[7] J. Yoon, V. Raghavan and Venu Chakilam Bitmap Indexing-based clustering and Retrieval of XML documents.

[8] J. Yoon, V. Raghavan and Venu Chakilam, BitCube: A Three Dimensional Bitmap Indexing for XML Documents, 13th International Conference on Scientific and Statistical Database Management, FairFax, VA, 2001.

[9] J. Yoon, Lafayette Presto authorization: a bitmap indexing scheme for high-speed access control to XML documents, IEEE Transactions on Knowledge and Data Engineering, Volume 18, Issue 7 pp 971-987, 2006

Multiscale Contrast Enhancement for Compressed Digital Images with Block Artifacts Consideration

Kashif Iqbal*, Asmatullah Chaudhry*, Asifullah Khan**, and Ajmal Bangash*
*Faculty of Computer Science & Engineering,
Ghulam Ishaq Khan (GIK) Institute of Engineering Science & Technology, Swabi, Pakistan
**Department of Computer and Information Sciences,
Pakistan Institute of Engineering and Applied Sciences, Nilore,Islamabad, Pakistan
kashifiqb@yahoo.com

Abstract - A simple and efficient algorithm is presented for contrast enhancement, of JPEG compressed images, in the Discrete Cosine Transform (DCT) domain. The algorithm enhances the DCT coefficients in accordance with their band importance. Since uniformly modifying all the frequency bands causes block artifacts, therefore, the low frequency bands are dealt with differently than the rest of the frequency bands. As the enhancement is done in the decompression stage, compressibility of the original image is not affected. Subjective and objective tests performed on various images validate the concept of multiscale contrast enhancement.

KEYWORDS

Contrast Enhancement, Digital Image Enhancement, Discrete Cosine Transform (DCT) and JPEG Compression.

I. INTRODUCTION

Image enhancement is the name of modifying certain features of an image such that it gives better look to the human visual system (HVS). Contrast enhancement is one such feature among the various features of an image, modification of which can produce better view of the image. In most of the cases, contrast enhancement may play the preprocessing role for later processing steps of the image enhancement.

Several methods are available for image enhancement in the spatial domain. Global Histogram Equalization [1] enhances the contrast of an image by rescaling the original image so that the histogram of the enhanced image follows some desired form. Since it treats all regions of the image equally, it often yields poor local performance due to the minimum detail preservation and performs not that good in the local context [2]. Some of the techniques are based on local contrast measure [3], while Beghcladi proposed a contrast enhancement technique via edge detection [4].

Compressed domain techniques for image enhancement usually operate on the transform coefficients of the images that are compressed by various transform methods. These include α-rooting algorithm [5-6] and modified unsharp masking [2].

J.Tang et. Al [7] presented a measure in the DCT domain for the contrast enhancement of JPEG compressed images. The contrast enhancement is done at the decompression stage which preserves the compressibility of the original image. Though this method improves the contrast of the image very well, it does not take into account the blocking artifacts that might appear due to the uniform modification of all the DCT bands i.e. it manipulates DCT coefficients using a single measure without considering the sensitivity of HVS to low-frequencies.

In this paper, we modify the DCT coefficients according to their band importance, instead of modifying them uniformly, for contrast enhancement of the digital images in compressed domain. The contrast enhancement is improved by giving more importance to the coefficients in the higher frequency bands.

This paper is organized as follows: Section 2 introduces JPEG image compression scheme. Section 3 describes contrast enhancement in DCT domain, the used algorithm and the proposed modification for contrast enhancement. Performance of the proposed scheme is given in section 4 and section 5 concludes the paper.

II. JPEG – DCT BASED IMAGE COMPRESSION

In this section, we present a brief review of JPEG, a DCT-based image compression technique. The idea of JPEG can easily be extended to other DCT-based image compression standards, such as MPEG 2, and H.26X.

A. JPEG Compression

A JPEG compression system consists of two parts, an Encoder and a Decoder. In the encoder, the image is partitioned into non-overlapping blocks of 8x8 pixels. Then DCT is performed on each 8x8 block, followed by the quantization of these DCT coefficients. Finally, the quantized coefficients are zig-zag scanned and entropy encoded.

In order to reconstruct the image, the decoder decodes the compressed image using entropy decoding. Dequantization is performed by point wise multiplication with the quantization table, and then inverse DCT-Transform is applied to obtain the image back.

The two-dimensional DCT of an 8 x 8 block $I_{j,k}$ in an original image is given by:

$$F(u,v)=\frac{1}{4}C(u)C(v)\sum_{j=0}^{7}\sum_{k=0}^{7}I(j,k)\cos\left\{\frac{\pi}{8}[u(j+\frac{1}{2})]\right\}\cos\left\{\frac{\pi}{8}[v(k+\frac{1}{2})]\right\}$$

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 267–272.

where $u,v = 0, 1, \ldots, 7$, and

$$C(n) = \begin{cases} 1/\sqrt{2} & for \;\; n = 0 \\ 1, & otherwise \end{cases}$$

The inverse DCT transformation can be written as:

$$I(j,k) = \frac{1}{4} C(u)C(v) \sum_{j=0}^{7} \sum_{k=0}^{7} F(u,v) \cos\left\{\frac{\pi}{8}[u(j+\frac{1}{2})]\right\} \cos\left\{\frac{\pi}{8}[v(k+\frac{1}{2})]\right\}$$

Here $j, k = 0, 1 \ldots 7$.

The coefficient F_{00} in the upper left corner of an 8 x 8 DCT block is called the "DC Coefficient", and the other 63 elements are called "AC Coefficients". The DC coefficient contributes for the average value of a single block in the original image, while the AC coefficients depend on the variations in gray levels.

The DCT coefficients in each frequency band vector have approximately similar spatial frequency properties. If only one frequency band is selected and an image block is reconstructed using the selected band, the reconstructed block can be thought of as bandpass version of the original block with frequency n, where $n = u + v$. The frequency contents of the bandpass image block correspond with higher frequencies as the band number increases.

$$\begin{bmatrix} F_{00} & F_{01} & F_{02} & F_{03} & F_{04} & F_{05} & F_{06} & F_{07} \\ F_{10} & F_{11} & F_{12} & F_{13} & F_{14} & F_{15} & F_{16} & F_{17} \\ F_{20} & F_{21} & F_{22} & F_{23} & F_{24} & F_{25} & F_{26} & F_{27} \\ F_{30} & F_{31} & F_{32} & F_{33} & F_{34} & F_{35} & F_{36} & F_{37} \\ F_{40} & F_{41} & F_{42} & F_{43} & F_{44} & F_{45} & F_{46} & F_{47} \\ F_{50} & F_{51} & F_{52} & F_{53} & F_{54} & F_{55} & F_{56} & F_{57} \\ F_{60} & F_{61} & F_{62} & F_{63} & F_{64} & F_{65} & F_{66} & F_{67} \\ F_{70} & F_{71} & F_{72} & F_{73} & F_{74} & F_{75} & F_{76} & F_{77} \end{bmatrix}$$

Fig. 1. 8x8 DCT output block

III. CONTRAST ENHANCEMENT IN DCT DOMAIN

Our proposed image enhancement algorithm is based on the contrast measure presented by J.Tang. We first review that contrast enhancement and then present our modification in the method.

A. *Contrast Measure*

The HVS detects the details of the scene based on the ratio between high-frequency and low-frequency contents. Therefore, the spatial frequency characteristics can be used to define a relative measure of scene contents. For this purpose, the 64 coefficients in an 8 x 8 DCT block are classified into 15 frequency bands, as shown in Fig.1**.** The first band consists of the dc coefficient F_{00}, second band contains elements F_{01} and F_{10}, third band consists of F_{02}, F_{11} and F_{20} and so on. The number of elements in each spectral band can be obtained by:

$$N = \begin{cases} k+1, & k < 8 \\ 15-k, & k \geq 8 \end{cases} \tag{1}$$

The local contrast measure of J. Tang is defined on each band except band number 0 (F_{00}). The contrast at *nth* band is defined as:

$$c_n = \frac{E_n}{\sum_{k=0}^{n-1} E_k} \tag{2}$$

Where

$$E_k = \frac{\sum_{u+v=k} |F(u,v)|}{N} \tag{3}$$

is the average amplitude over a spectral band.

This contrast measure c_n in the *nth* band is the ratio of the frequency content of the bandpass image block obtained by the *nth* band to that of the lowpass image block. Let the contrast of the original block be $C = (c_1, c_2, \ldots, c_{14})$, where c_n is the contrast at the *nth* frequency band corresponding to E_n, and let the contrast of the enhanced block be denoted by $\overline{C} = (\overline{c}_1, \overline{c}_2, \ldots, \overline{c}_{14})$.

The method presented by J.Tang enhances the contrast uniformly for all frequency bands as:

$$\overline{c}_n = \lambda . c_n \tag{4}$$

B. *Proposed Modification*

Our objective of contrast enhancement has the following two purposes:

1. To effectively enhance the contrast of the image in such a way as to avoid block artifacts in the resultant image.
2. To enhance the image details by modifying the DCT coefficients according to their band importance.

This leads us to define a multiscale contrast enhancement factor that not only avoids block artifacts in the image, but also enhances the details of the image.

Eq. (4) can thus be modified as follows:

$$\overline{c}_n = \lambda_n c_n \tag{5}$$

resulting in:

$$\frac{\lambda_n E_n}{\sum_{k=0}^{n-1} E_k} = \overline{c}_n = \lambda_n c_n = \frac{\overline{E}_n}{\sum_{k=0}^{n-1} \overline{E}_k} \tag{6}$$

where λ_n is the contrast enhancement factor for the *nth* band. Eq. (6) can be formulated as:

$$\overline{E}_n = \lambda_n H_n E_n, \quad n \geq 1 \tag{7}$$

where

$$H_n = \frac{\sum_{k=0}^{n-1} \overline{E}_k}{\sum_{k=0}^{n-1} E_k}, \quad n \geq 1 \tag{8}$$

Here, $E_{00} = \overline{E}_{00} = | F_{00} |$ and H_n can be obtained by recursion. Finally the enhanced DCT coefficients $\overline{F}_{u,v}$ are obtained using Eq. (7) as:

$$F_{u,v} = \lambda_{u+v} H_{u+v} F_{u,v}, \quad u + v \geq 1 \tag{9}$$

C. Deciding Contrast Enhancement Value

The contrast enhancement λ_n in our proposed scheme depends on two factors. First, the coefficients of bands 1 and 2 (not the dc-coefficient) are extremely low frequency components and cause blocking artifacts if modified too much. To avoid blocking, these two bands should be treated separately than the other bands. The second is based on the exploitation of the HVS for adjusting the enhancement factor in such a way that more weightage is given to the DCT coefficients in high frequency bands.

This discussion leads the enhancement factor to be defined as:

$$\lambda = \begin{cases} \lambda_1, & for\, 1 \leq n \leq 2 \\ \lambda_n, & for\, 2 < n \leq 14 \end{cases} \tag{10}$$

To decide for λ_1 and λ_n, we start with a smaller value than that of J.Tang. The minimum value of λ_1, that avoids block artifacts, has been found to be 1.1. For λ_n, we simulated its values to be a function, as:

$$\lambda_n = \lambda_{(n-1)} + \alpha, \; 3 < n \leq 14 \tag{11}$$

where the starting value of λ for $n = 3$ has been set empirically as 2.0, as shown in Fig. 2, and α gives a linear growth to the enhancement factor λ. α should be chosen such as it does not produce noise effects in the higher frequencies.

IV. ANALYSIS OF THE PROPOSED SCHEME

This section contains a series of experiments to demonstrate the performance of our proposed modification. The results are also compared with two conventional methods.

A. Existing Methods

Two algorithms are implemented and compared with our proposed method as base line schemes. First method is Global Histogram Equalization, which modifies the contrast of an image by altering the image such that its intensity histogram has a desired shape [2]. By comparing Global Histogram Equalization method with our approach, it is observed that Global Histogram Equalization over-enhanced the images. Some regions of the resultant image appear deep darkened and others deep brightened (see fig. 3(b)). The second conventional approach used for comparison is that of J. Tang (see fig. 3(c)).

B. Enhancement Quality Metrics

For quantitative evaluation of our proposed scheme, we use various quality metrics. Wang et al [8] presented structural similarity index measure (SSIM) for image quality assessment that is based on human visual system. The main function of the HVS is to extract structural information from the viewing field. Therefore, a measurement of structural information loss can provide a good approximation to perceived image distortion.

Other than that, we have used weighted peak signal to noise ratio (wPSNR) as measure of the image quality.

C. Experimental Results and Discussions

The proposed algorithm has been tested on various images. It is observed that the contrast of the resultant image is not only enhanced, but also no or less block artifacts are observed as shown in Fig. 3(d). Table 1 shows the quantitative results in terms of SSIM. It can be seen that regardless of the enhancement made by our algorithm, the structural similarity index of our approach is better than that of J. Tang and Global Histogram Equalization approaches. Similarly, wPSNR is depicted in Table 2. It again shows that our proposed method outperforms Histogram Equalization and J. Tang Method.

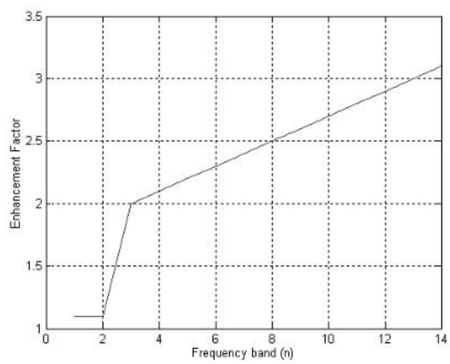


Fig. 2. Enhancement factor vs band number in 8x8 DCT block

TABLE I
PERFORMANCE COMPARISON IN TERMS OF SSIM

Images	Global Histogram Equalization	J. Tang Method	The Proposed Approach
Lena	0.8972	0.6869	0.9443
Wheel	0.3961	0.4441	0.8900
Barber	0.7156	0.8237	0.8689
Clock	0.6363	0.8422	0.8942
Moon Surface	0.5469	0.8706	0.8980

Fig. 3. (a) Decompressed JPEG Lena image (b) Global Histogram Equalization method (c) J.Tang's method (d) The proposed approach

TABLE II
PERFORMANCE COMPARISON IN TERMS OF WPSNR

Images	Global Histogram Equalization	J. Tang Method	The Proposed Approach
Lena	18.37	15.15	22.28
Wheel	12.06	16.85	24.94
Barber	18.28	17.95	20.21
Clock	13.03	12.38	13.50
Moon Surface	16.46	18.23	19.41

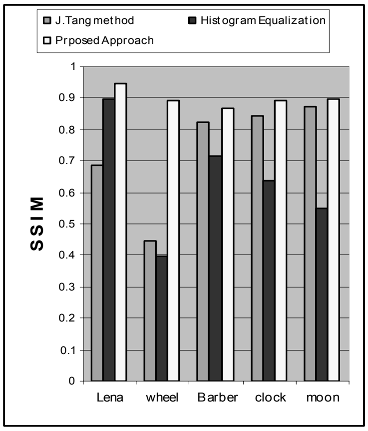
Fig. 4 illustrates the results of both tables graphically. Similarly, Fig. 5, 6 and 7 shows that the proposed algorithm enhances the images without causing any block artifacts.

V. CONCLUSION

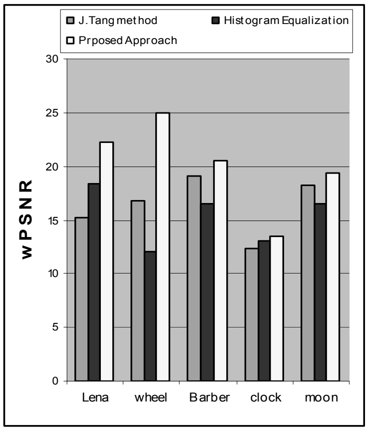
In this paper, we have developed an image enhancement algorithm, for compressed images, in DCT domain. The proposed method efficiently enhances the contrast of the resultant image, keeping in view reducing blocking artifacts that arise due to the uniform modification of all the frequency bands. Moreover, it takes into account the role of higher frequency coefficients in the contrast enhancement of the image. Objective tests using various images validate the concept of our proposed method.

ACKNOWLEDGMENT

The authors would like to thank National Engineering and Scientific Commission (NESCOM) and Higher Education Commission for their financial support and GIK Institute of Engineering Sciences & Technology for facilitating this research.



(a)

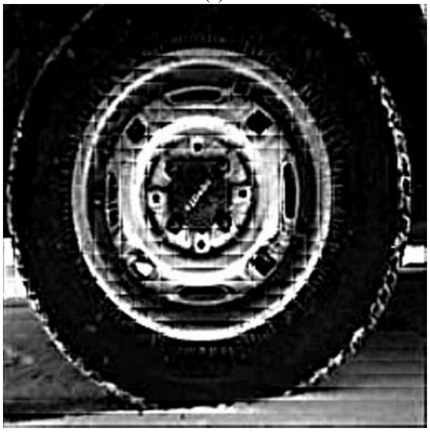


(b)

Fig. 4. Plots of (a) SSIM (b) wPSNR

(a)

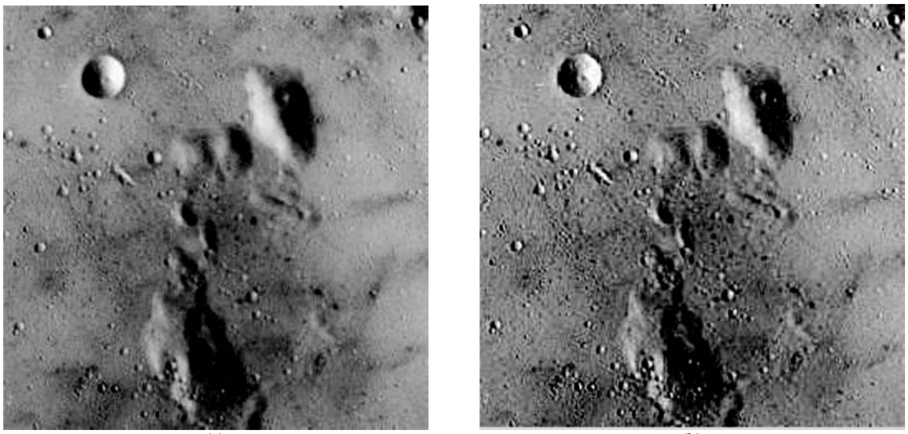
(b)



(c)

(d)

Fig. 5. (a) Decompressed JPEG Wheel image (b) Global Histogram Equalization method (c) J.Tang's method (d) The proposed approach



(a) (b)

Fig.6. (a) Decompressed JPEG Moon Surface image (b) The proposed approach

(a) (b)

Fig.7. (a) Decompressed JPEG Barber image (b) The proposed approach

REFERENCES

[1]. R. Gonzalez and P. Wintz, "Digital Image Processing", MA: Addison-Wesley, 1987

[2]. A. Polesel, G. Ramponi, and V.J. Mathews, "Image enhancement via adaptive unsharp masking", *IEEE Trans. Image Processing*, vol. 9, pp. 505-510, Mar. 2000.

[3]. Ji, T.L., Sundareshan, M.K., Roehrig, H., 1994. Adaptive image contrast enhancement based on human visual properties. IEEE Trans. Med. Imaging 13 Dec, 573-586.

[4]. Beghcladi, A., Negrate, A.L., 1989. Contrast enhancement based on local detection of edges. Computer Vission Graphics Image Processing. 46, 162-174.

[5]. S.S. Agaian, K. Panetta, and A. M. Grigoryan, "Transform-based image enhancement algorithms with performance measure," IEEE Trans. Image Processing, vol. 10, pp. 37-382, Mar. 2001.

[6]. Peli, E., 1990. Contrast in complex images. J. Opt. Soc. Amer. A, vol. 7, pp. 2032-2040, 1990.

[7]. J. Tang, E. Peli, and S. Acton, "Image enhancement using a contrast measure in the compressed domain", *IEEE Signal Processing. Lett.* 10(10), 289-292, 2003.

[8]. Z.Wang, A.C. Bovic and H.R. Sheikh, "Image Quality Assessment: From Error Visibility to Structural Similarity", *IEEE Transactions on Image Processing,* Vol. 13, No. 4, April 2004.

An Aspect-Oriented Model to Monitor Misuse

K. Padayachee
School of Computing, University of South Africa
Pretoria, 0003 South Africa

J.H.P. Eloff
Department of Computer Science, University of Pretoria
Pretoria, 0002 South Africa

Abstract- **The efficacy of the aspect-oriented paradigm has been well established within several areas of software security as aspect-orientation facilitates the abstraction of these security-related tasks to reduce code complexity. The aim of this paper is to demonstrate that aspect-orientation may be used to monitor the information flows between objects in a system for the purposes of misuse detection. Misuse detection involves identifying behavior that is close to some previously defined pattern signature of a known intrusion.**

I. INTRODUCTION

The efficacy of the aspect-oriented paradigm has been well established within several areas of security, such as authentication, access control, encryption and software tampering. Typically security concerns tend to crosscut objects, resulting in code tangling. However the aspect-oriented paradigm facilitates the abstraction of these security-related tasks to reduce code complexity. Crosscutting concerns are related issues that are scattered throughout the functionality of an application [1]. The aim of this paper is to demonstrate that aspect-orientation may be used to monitor the information flows between objects in a system for the purposes of misuse detection. Misuse-detection involves identifying behavior that is comparable to some previously defined pattern signature of a known intrusion.

Application-level bugs are often exploited to compromise the security of a system and it is vital to correct these vulnerabilities instantaneously. For instance, 'design-level problems accounted for about 50% of the security flaws uncovered during Microsoft's "security push" in 2002. Sufficient protection of software applications from attacks, however, is beyond the capabilities of network and operating system-level security approaches (e.g. cryptography, firewall and intrusion detection) because they lack knowledge of application semantics' [2].

Detecting programming attacks should ideally follow an approach similar to the course of action advocated by Newsome and Song [3]: A detection mechanism should detect unknown attacks early, before the system is compromised. Secondly, once a new exploit attack is detected, attack signatures must be developed that can be used to filter out those attacks efficiently until the vulnerability can be patched. This paper focuses on the latter event, when an attack is known and where an interim measure must be instituted until the problem is resolved.

The use of aspect-orientation in information security has been validated by several studies ([4-6]). Aspect-orientation promotes reusability since a security aspect may be reused for other applications [1]. For example, access control has similar requirements for most applications. Vanhaute and De Win [7] have derived reusable generic aspects from typical security concerns. As aspectual components do not need to have hard-wired names of objects, an aspect may be easily reused [8]. Furthermore, the aspect-oriented paradigm is highly extensible as it is flexible enough to accommodate the implementation of additional security features after the functional system has been developed, as crosscutting concerns may be added or removed without making invasive modifications to original programs [9].

This paper examines the strategy of using the aspect-oriented paradigm to reveal patterns of information flow to detect programming attacks. The first two sections explore the concepts of misuse detection systems and information flow control respectively, while the discourse of the subsequent two sections focuses on aspect-oriented programming and its influence on software security. Section 6 demonstrates how an aspect-oriented methodology may be used to detect information flow patterns that signify programming attacks. Sections 7 and 8 conclude with directions for future work and insights gathered from the experiment conducted.

II. MISUSE DETECTION SYSTEMS

Anomaly detection relies on identifying all behavior that is abnormal for an entity. While misuse detection involves flagging behavior that is close to some previously defined pattern signature of a known intrusion. The disadvantage of the first approach is that it does not necessarily detect undesirable behavior, and that the false alarm rates can be high. The problem with tracking all information flows would be the difficulty in identifying an anomaly as these logs will probably be large. The problem with misuse-based detection is that the anomaly must be known in advance.

While misuse-based detection cannot detect new intrusions, in actual systems, anomaly detection systems have the advantage of detecting previously unknown intrusions[10]. Anomaly detection methods involve various machine learning and statistical techniques [11]. This paper will not examine the issues surrounding pattern recognition.

Recently there has been a trend towards using hybrid frameworks combining both misuse detection and anomaly detection components which, in effect, reduces the inefficiencies and maximizes the strengths of both techniques (see [12]). The other significant trend is the movement towards the inclusion of intrusion detection systems on the application-level. Most intrusion detection

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 273–278.

systems are essentially based at the network-level or operating system level. It has been noted recently that there is a need to consider the application-level, in terms of monitoring the interaction between the user and the application [13]. 'Application-level bugs are more frequent than kernel-level bugs and, therefore, applications are often the means to compromise the security of a system. Detecting these attacks can be difficult, especially in the case of attacks that exploit application-logic errors' [14]. However 'to detect attacks exploiting application-logic errors, it is desirable to be able to perform selective, application specific auditing in certain points of the application's control flow. The problem is that few applications provide hooks for instrumenting [sic] their control flows, and, even if these hooks are available, they may not be in the right places. In addition, the instrumentation technique would be application-specific and not easily portable to different applications' [14]. It is evident that aspect-orientation may be ideal for providing these 'hooks' through the use of pointcut designators [15].

Another newly identified trend is the use of information flow control to support misuse detection. In this research attempts are made to find a solution within the aspect-oriented paradigm while incorporating some of these trends. The model presented here, is based on misuse detection within the application-level using information flow control analysis.

III. Background On Information Flow Control

Information is exchanged among variables in procedural programs and by messages in object-oriented systems. An illegal flow arises when information is transmitted from one object to another object in violation of the information flow security policy [16]. A transfer of information does not necessarily occur every time a message is passed. An object acquires information by changing its internal state, as a result of changing the values of some of its attributes. Thus, if no such changes occur as a result of a message invocation in response to a message, then no information has been transferred [17]. There have been two basic types of information flow controls available within the object-oriented perspective, namely language-based information flow controls [18] and information flow controls based on message filtering [16, 19].

Language-based information flow controls are enforced through the use of security-typed languages where program variables and expressions are augmented with annotations that specify policies on the use of the typed data. Language-based information-flow techniques necessitates that the programmer must not only understand the algorithm to be implemented but must also understand what the desired security policy is and how to formalize it using annotations [20]. Further security policies may not be available during functional design, thereby resulting in inconsistencies. The aspect-oriented paradigm enables security policies to be separated from the code and accordingly security policies may be coded independently of other requirements [21]. In general, information models are difficult to implement. Hence the message filtering model developed by Jajodia and Kogan [19] considers only primitive operations such as *read* and *write* methods. As the aspect-oriented paradigm facilitates genericity through the use of wildcards, it may extend the message filter model beyond considering only primitive operations [22].

The methodologies presented above, are solely based on preventing illegal information flow. However, Masri and Podgurski [23] presented a novel approach to detect attacks against application software using dynamic information flow analysis. Where certain patterns of information flow may be used to detect vulnerabilities and possible attacks. This paper shares this notion but surveys an aspect-orientation implementation of information flow, as an alternative technology. Due to the genericity offered by aspect-orientation, the model presented here may be used within other contexts during security risk analysis, where the illumination of specific information flows to detect vulnerabilities is required.

IV. Background On Aspect-Oriented Programming

In every object-oriented software design there are core concerns. In a robotic system, for instance, these concerns involve motion management and path computation. The concerns are located in a particular scope and are not required in any other scope. Other concerns are common to many of a system's modules like logging, authorization and persistence. These system-wide concerns are called 'crosscutting concerns' and the re-implementation of one issue in different modules is called 'code scattering' [24]. Aspect-oriented programming addresses the problem of code scattering by localizing these crosscutting concerns into a modular unit called an aspect.

An aspect is a modular unit of a crosscutting implementation that is provided in terms of pointcuts and advices, specifying what (advice) and when (pointcut) its code is going to be executed [25]. In terms of codification, aspects are similar to objects. However, aspects observe objects and react to their behavior [26]. An aspect is a piece of code that describes a recurring property of a program and can span multiple classes, interfaces or aspects [8]. Unlike a class though, aspects are injected into other types. Aspects improve the separation of concerns by making it possible to cleanly localize crosscutting design concerns. They also allow programmers to write, view and edit a crosscutting concern as a separate entity.

During program execution, there will be certain well-defined points where calls to aspect code would be inserted [25]. These are known as join points. Aspects introduce their supplemental functionality at these join points [26]. A pointcut is a set of join points described by a pointcut expression. An advice declaration is used to specify code that should run when the join points specified by the pointcut expression are reached [27]. The advice code will be executed when a join point is reached, either before or after the execution proceeds. For example, AspectJ supports *before*, *after* and *around* advices, depending on the time the code is executed [28]. A *before* (*after*) advice on a method execution defines code to be run before (after) the particular method is actually executed. An *around* advice defines code which is executed when the join point is reached and has control over whether the computation at the join point (i.e. an application method) is allowed to be executed or not [29] .Combining the application functional code and its specific aspects generates the final application. These two entities will be combined at compile time by invoking a special tool called a 'weaver' [8].

V. RELATED WORK ON APPLYING THE CONCEPT OF ASPECT-ORIENTED SECURITY IN CREATING SECURE SOFTWARE

Security is often extracted as a separable concern, due to its orthogonal nature in respect of the functional requirements of a system, hence the separation-of-concerns principle of the aspect-oriented paradigm is suited to addressing security concerns [30]. Aspect-oriented software development is relevant for all major pillars of security: authentication, access control, integrity, non-repudiation, as well as for the supporting administration and monitoring disciplines required for effective security [31]. Even security-related bugs such as buffer overflows or race conditions can be considered a security-related concern [6]. Security aspects can be used to modularize access control and authentication (see [6], [32] and [33]).

In a study related to information flow control, Masuhara and Kawauchi [34] found there was no possible way to define a pointcut that would be able to detect whether a string was from an unauthorized source or not or contained unwanted information. Hence they proposed a new pointcut called *dflow* that addresses the dataflow between join points as an extension to the AspectJ Language. Although this study is related to information flow, the authors do not address security classifications and their dataflow definition ‘only deals with direct information flow’. Further, they do not comment on the propagation of information among objects in a system. To address these shortcomings, we have conducted case-studies to demonstrate that the aspect-oriented methodology might be useful for detecting illegal flows between objects [22].

These experiments showed that aspects could be utilized to identify flows between objects. For instance, this aspect’s advice could decide, upon examining the given message and classification of the sender and receiver, whether to permit the information flow or not. In an unpublished work by Padayachee, Eloff and Bishop [35], this model was further generalized, so that it may be used in other contexts as well. The *Flow* aspect considered those actions that resulted in an attribute being assigned (set) or returned from an object. This notion actually addresses all interactions between objects, including when objects are being instantiated. When an attribute is returned from a message, the reference of this attribute is stored in an appropriate container. When an attribute in an object is being assigned to a particular value, this container is inspected to check if the value was obtained from another object. If this value was obtained from another object then an appropriate action may be taken if the information flowing should not be permitted according to the information flow policy (see Fig. 1 below).

```
public aspect Flow {
pointcut getMethods():  get(* *.*);
before( )returning(Object x): getMethods() &&
(within (A || B || C //...and other objects being tracked )){
//Information flowing out of this object
//Store information about this joinpoint – Such as the reference of
//this object.
}
pointcut setMethods(Object x):  ((set (* *.*) || call (* *(..))) && args(x)
&& within( A || B || C //...and other objects being tracked ));
before(Object x): setMethods(x) {
//if there is an illegal flow between the sender and receiver object
//  Respond appropriately
//else
//  Allow Process to proceed as normal
}
}
```

Fig. 1. Aspect to determine the flow of information between objects.

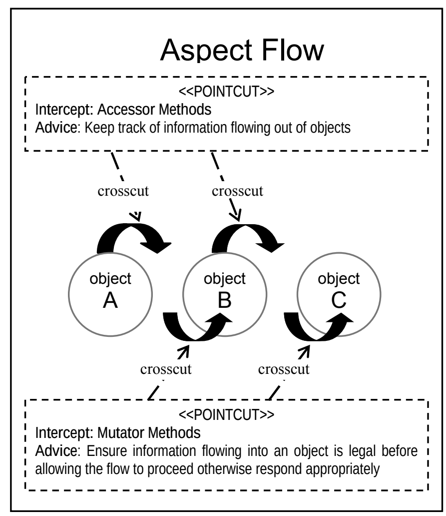


Fig. 2. Aspect Flow intercepts information flow between interacting objects.

For instance, this aspect’s advice could decide, upon examining the given message and classification of the sender and receiver, whether to permit the information flow or not. Figure 2 illustrates - when sender object (A) sends information to receiver object (B), this flow is intercepted and tested if it violates the specified information flow policy. If the flow does not disobey the specified information flow policy then the aspect allows the flow to proceed to receiver object (B), otherwise the aspect does not allow the flow to occur or performs a specific action. The model however, has limitations as it only identifies explicit flows [35]. We have also demonstrated via a case study that using aspect-oriented flow model may useful in identifying programming attacks [22]. In this paper we reprised this model in a more generalized context such that it may be useful be in identifying programming attacks.

VI. Case Study To Demonstrate The Practicality Of The Model

With respect to the aspect-oriented flow model described above, a small system was built based on the example provided by Masri and Podgurski [23] to test the possibility that such system can be fully implemented. The system was built using Aspect J (ajdt_1.2_for_eclipse_3.0) as an extension to Java (J2SDK1.4.2_05), and the Eclipse 3.0 IDE. We wanted to show that aspects could be used to detect vulnerabilities. To this end, we considered a server application comprising of three classes `Server`, `Session` and `Account` (depicted in Fig. 3), where there was vulnerability in that the server allows a malicious client to avoid getting charged for his/her connection time:

1) Attacker opens a first session (session1) and uses it for a long time.
2) Attacker opens a second session (session2)
3) Attacker closes session1 (immediately after step 2)
4) Attacker closes session 2 (immediately after step 3)

This attack basically induces the following information flows:

clientAccount => Session1
ClientAccount => Session2
Session1 => ClientAccount
Session2 => ClientAccount

This experiment sought to replicate this vulnerability by using the aspect-oriented information flow model to identify patterns of flow that indicate misuse.

```
//______________________________________________________
public class Account {
 public Account (String user){
     name = user;
 }
 public Double getCredit() {
     return credit;
}
 public void setCredit(Double credit){
     this.credit = credit;
}
 public String name;
 private Double credit  = new Double (3600000);//1 hour
}
public class Session {
 Session(Double startCredit){
     this.startCredit = startCredit;
     startTime =  new Double (System.currentTimeMillis());}
 public Double computeCredit(){
return new Double (startCredit.doubleValue() -
(System.currentTimeMillis() -
 (startTime).doubleValue()));
}
 private Double startCredit ;
 private Double startTime;
}
public class Server {
 Session openSession(String user){
     theAccount = new Account(user);
     Account account = theAccount;
     return new Session(account.getCredit());
}
 void closeSession(Session session){
     Account account = theAccount;
     account.setCredit( (session.computeCredit()));
}
 Account theAccount;
}
public class SimpleAttack {
 public void accessServices(Server s, Session session, Double
Duration){
     // Do necessary action
 }
 public static void main(String[] args) {
     SimpleAttack s = new SimpleAttack();
     Double shortDuration = new Double (10);
     Double longDuration = new Double (1000);
     Server server = new Server();
     //OPEN SESSION 1
     Session session1 = server.openSession("User01");
     s.accessServices(server, session1,longDuration);
     //OPEN SESSION 2
     Session session2 = server.openSession("User01");
     //CLOSE SESSION 1
     server.closeSession(session1);
     //USE SESSION2 FOR A SHORT PERIOD.
     s.accessServices(server,session2,shortDuration);
     //CLOSE SESSION 2
     server.closeSession(session2);
 }
}
//______________________________________________________
```

Fig. 3. Defective server implementation and simple attack. Note only code relevant to the vulnerability/attack is shown. (adapted from [23]).

After the classes were developed and tested, the generalized aspect *Flow* (Fig.4.) was contextualized to identify this type of programming attack. The specific flows between the `Account` class and the `Session` class which were woven into the rest of the system were identified. The specifications involved adjusting the `getMethods()` pointcut to include the `ComputeCredit()` method calls. The *Flow* aspect only identifies flows when an object's data members flow out of an object. As the `ComputeCredit()` method did not involve a data member being returned, it had to be specifically named. The next issue involved identifying the objects involved. As the `Account` class had a name data member, this class did not pose a problem. However, as the `Session` class did not contain a data member that could be used to identify the object, an aspect `AddToSession`, was created specifically for that purpose.

```
//______________________________________________________
public aspect Flow {
private static Vector References = new Vector();
private static Vector JoinPoint_String =  new Vector();
private static int Count = 0;
pointcut getMethods(): get (* *.*) || execution(Double
Session.computeCredit(..) );
before () returning (Object x):getMethods() && (within (Session
|| Account))
{//Store info about this Join Point
 JoinPoint_String.add(Count, thisJoinPoint.getThis());
 References.add(Count,x);
 Count++;
}
pointcut setMethods(Object x): (set (* *.*) ) && args(x) &&
within (Session||Account);
before (Object x): setMethods(x)
{String Signature = thisJoinPoint.toString();
```

```
 for (int i = 0; i < Count; i++){
 if (References.get(i) == x){
System.out.print("Information Flowing From ");
 String PrevSignature = JoinPoint_String.get(i).toString();
 if (PrevSignature.indexOf("Session") > 0 ){
 System.out.print("SESSION: " + ((Session)
 JoinPoint_String.get(i)).myname);
 }
 else if (PrevSignature.indexOf("Account") > 0){
 System.out.print("ACCOUNT: " + ((Account)
 JoinPoint_String.get(i)).name);
 }
 if (Signature.indexOf("Session") > 0 ){
 System.out.println(" to SESSION: "
 +((Session)thisJoinPoint.getThis()).myname);
 }
 else if (Signature.indexOf("Account") > 0){
 System.out.println(" to ACCOUNT: "+
 ((Account)thisJoinPoint.getThis()).name);
 }
}}}
}
public aspect AddToSession {
 private static int Session.Count = 0;
 public int Session.myname = Count;
 pointcut SessionConstructor(): execution(Session.new(..));
 before(): SessionConstructor() {
 Session.Count++; ((Session) thisJoinPoint.getThis()).myname =
 Session.Count;}
}//
```

Fig. 4. Aspect-oriented implementation to identify patterns of misuse.

The following output (Fig.5.) was produced after the aspects (in Fig. 4.) were woven together with the classes given in Fig.3 using AspectJ. This output reflects the pattern of flow identified by Masri and Podgurski [23] which depicts a user exploiting a program vulnerability.

```
Information Flowing From ACCOUNT: User01 to SESSION: 1
Information Flowing From ACCOUNT: User01 to SESSION: 2
Information Flowing From SESSION: 1 to ACCOUNT: User01
Information Flowing From SESSION: 2 to ACCOUNT: User01
```

Fig. 5. The output produced reflects patterns of misuse detection.

There were a few limitations to this experiment. A few liberties were taken in adapting the aspect to identify the specified flows. It is not the most efficient solution but it rather reflects a 'hacking style' technique. The experiment sought to reveal how quickly the model given could be adapted to resolve the problem at hand. The model proposes that once vulnerability is discovered, an aspect can be used to monitor this vulnerability until a fix or patch is created. Thus this particular monitoring aspect is temporal and will be removed once the program is fixed. In this case, a single programmer was able to complete the task of adapting the aspect in an hour after the classes were built and tested. Several insights were gathered from this experiment. The limitations of AspectJ posed a problem in identifying the objects. It is easy to identify the classes used by referencing a special variable called *thisJoinPoint*, which contains reflective information about the current join point. However, it is difficult to determine the name of the object itself. It would be ideal if the *thisJoinPoint* variable could be expanded to resolve this issue.

A security risk analyst has to trace and track vulnerabilities in a system and this is difficult as he has to relate to the subtleties of the system that are understood by the designer and implementer [36]. Certainly using the model developed above can expose these subtleties without the security risk analyst understanding every program path in the program and constraints may be formed without relying on the programmers to abide by these constraints.

VII. Future Work

We presented an aspect-oriented flow model to identify patterns of information flow. We tested the model on a very small system. In the future a thorough case-study approach would be taken on a more scalable solution to access the flexibility and usability of the model. Although the model was able to produce the flow pattern, the semblance of the pattern reproduced here was discerned manually. A future undertaking may involve developing a tool to automate the process.

VIII. Conclusion

Despite its appeal, information flow mechanisms are difficult to manage in practice and require programmers to be security experts. The aspect-oriented paradigm can be used to add security to existing systems and due to the separation of roles between application developers and security, it can make the management of security polices easier. In this paper, an information flow control model to detect misuse using aspect-orientation is posited. This model may be useful to security risk analysts for identifying security vulnerabilities. Aspects offer several benefits in terms of compact code and increased confidence, but there could be drawbacks as it is a new technology.

References

[1] B. De Win, B. Vanhaute and B. Decker, "Security through Aspect-Oriented Programming," in *Advances in Network and Distributed Systems Security, IFIP TC11 WG11.4 First Working Conference on Network Security*, November 2001, pp.125-138.

[2] D. Xu and K. Nygard, "A Threat-Driven Approach to Modeling and Verifying Secure Software," in *Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering*, November 2005, pp.342-346.

[3] J. Newsome and D. Song, "Dynamic Taint Analysis for Automatic Detection, Analysis, and Signature Generation of Exploits on Commodity Software," in *12th Annual Network and Distributed System Security Symposium*, February 2005, pp.1-17.

[4] P. Falcarin, M. Baldi and D. Mazzochi, "Software Tampering Detection Using AOP and Mobile Code," presented at *International Conference on Aspect-Oriented Software Development (AOSD'04)*, Lancaster, UK, 2004.

[5] G. Boström, "A Case Study on Estimating the Software Engineering Properties of Implementing Database Encryption as an Aspect," presented at *Proceedings of the 3rd international conference on Aspect-oriented software development*, Lancaster, UK, 2004.

[6] B. De Win, W. Joosen and F. Piessens, "Aosd Security: A Practical Assessment," presented at *Workshop on Software engineering Properties of Languages for Aspect Technologies (SPLAT03)*, Boston, Massachusetts, 2003.

[7] B. Vanhaute and B. De Win, "AOP, Security and Genericity," presented at *1st Belgian AOSD Workshop*, Vrije Universiteit Brussel, Brussels, Belgium, 2001.

[8] J.P. Choi, "Aspect-Oriented Programming with Enterprise Javabeans," in *Fourth International Enterprise Distributed Object Computing Conference (EDOC'00)*, September 2000, pp.252-261.

[9] N. Ubayashi, H. Masuhara and T. Tamai, "An AOP Implementation Framework for Extending Joint Point Models," presented at *ECOOP' 2004 Workshop on Reflection, AOP and Meta-Data for Software Evolution*, Oslo, Norway, 2004.

[10] R.A.V. Kemmerer, G., "Intrusion Detection: A Brief History and Overview," *Computer*, vol. 35, pp.27-30, April 2002.

[11] G. Adam, "Anomaly Detection in Distributed Computer Communication Systems," *Cybernetics and Systems*, vol. 37, pp.635-652, September 2006.

[12] J. Zhang and M. Zulkernine, "Hybrid Network Intrusion Detection Technique Using Random Forests," in *The First International Conference on Availability, Reliability and Security (ARES 2006)*, April 2006, pp.262-269.

[13] M. Suydam, "Application-Based Idss: Upping the Application Ante," 1999, http://infosecuritymag.techtarget.com/articles/1999/ids.shtml.

[14] J. Zhou and G. Vigna, "Detecting Attacks That Exploit Application-Logic Errors through Application-Level Auditing," in *Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC'04)*, December 2004, pp.168-178.

[15] E. Tombini, H. Debar, L. Me and M. Ducasse, "A Serial Combination of Anomaly and Misuse Idses Applied to Http Traffic," in *20th Annual Computer Security Applications Conference, 2004.*, December 2004, pp.428-437.

[16] P. Samarati, E. Bertino, A. Ciampichetti and S. Jajodia, "Information Flow Control in Object-Oriented Systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 9, pp.624-538, July-August 1997.

[17] S. Jajodia, B. Kogan and R. Sandhu, "A Multilevel-Secure Object-Oriented Data Model," Tech. Rep., pp.596-616, 1992.

[18] A. Sabelfeld and A.C. Myers, " Language-Based Information-Flow Security," *IEEE Journal on Selected Areas in Communications*, vol. 21, pp.5-19, January 2003.

[19] S. Jajodia and B. Kogan, "Integrating an Object-Oriented Data Model with Multilevel Security," in *1990 IEEE Symposium on Security and Privacy*, May 1990, pp.76-85.

[20] S. Zdancewic, "Challenges in Information-Flow Security," presented at *The First International Workshop on Programming Language Interference and Dependence (PLID)*, Verona, Italy, 2004.

[21] J. Viega, J.T. Bloch and P. Chandra, "Applying Aspect-Oriented Programming to Security," *Cutter IT Journal*, vol. 14, pp.31-39, February 2001.

[22] K. Padayachee and J.H.P. Eloff, "Information Flow Control within AOP," October 2006, http://www.osprey.unisa.ac.za/TechnicalReports/ index. html.

[23] W. Masri and A. Podgurski, "Using Dynamic Information Flow Analysis to Detect Attacks against Applications," presented at *Proceedings of the 2005 workshop on Software engineering for secure systems-building trustworthy applications*, ACM Press New York, NY, USA, St. Louis, Missouri, 2005.

[24] N. Kuntze, T. Rauch and A.U. Schmidt, "Security for Distributed Web Applications Via Aspect Oriented Security," presented at *Conference Information Security South Africa*, Sandton, South Africa, 2005.

[25] F. Ortin and J.M. Cueva, "Dynamic Adaptation of Application Aspects," *Journal of Systems and Software*, vol. 71, pp.229-243, May 2004.

[26] J. Viega and J. Voas, "Can Aspect-Oriented Programming Lead to More Reliable Software," *IEEE Software*, vol. 17, pp.19-21, November 2000.

[27] D. Mahrenholz, O. Spinczyk and W. Schröder-Preikschat, "Program Instrumentation for Debugging and Monitoring with Aspectc++," in *Proceedings of The 5th IEEE International Symposium on Object-oriented Real-time Distributed Computing*, April - May 2002, pp.249-256.

[28] B. De Win, W. Joosen and F. Piessens. "Developing Secure Applications through Aspect-Oriented Programming." in *Aspect-Oriented Software Development*, Aksit, M., Clarke, S., Elrad, T. and Filman, R.E. Eds., Boston: Addison-Wesley, 2002, pp.633–650.

[29] G. Kiczales, E. Hilsdale, J. Hugunin, M. Kersten and J. Palm, "Getting Started with Aspectj," *Communications of the ACM*, vol. 44, pp.59-65, October 2001.

[30] P. Robinson, M. Rits and R. Kilian-Kehr, "An Aspect of Application Security Management," presented at *AOSD'04 International Conference on Aspect-Oriented Software Development*, Lancaster, UK, 2004.

[31] R. Bodkin, "Enterprise Security Aspects," presented at *AOSD'04 International Conference on Aspect-Oriented Software Development*, Lancaster, UK, 2004.

[32] V. Shah and F. Hill, "An Aspect-Oriented Security Framework," in *DARPA Information Survivability Conference and Exposition*, April 2003, pp.143.

[33] P. Slowikowski and K. Zielinski, "Comparison Study of Aspect-Oriented and Container Managed Security," presented at *AAOS 2003:Analysis of Aspect-Oriented Software (Workshop held in conjunction with ECOOP 2003)*, AGH University of Science and Technology, Darmstadt, Germany, 2003.

[34] H. Masuhara and K. Kawauchi, "Dataflow Pointcut in Aspect-Oriented Programming," in *Proceedings of The First Asian Symposium on Programming Languages and Systems (APLAS'03)*, November 2003, pp.105-121.

[35] K. Padayachee, J.H.P. Eloff and J. Bishop, "Aspect-Oriented Information Flow Control," unpublished.

[36] H. Chivers and J. Jacob, "Specifying Information-Flow Controls," in *Proceedings of the 25th IEEE International Conference on Distributed Computing System Workshops (ICDCSW '05)*, June 2005, pp.114-120.

Ontology Development for E-Marketing Mix Model Mapping with Internet Consumers' Decision-Making Styles

Sam, K. M.
Affiliation: University of Macau,
Faculty of Business Administration, University of Macau, Av. Padre Tomás Pereira S.J., Taipa, Macao
&
P. Lei
School of Engineering, Design & Technology, University of Bradford, Richmond Road, Bradford BD7 1DP
&
Chatwin, C. R.
School of Engineering, University of Sussex, Falmer, Brighton BN1 9QT

***Abstract* - Based on a literature review of E-Marketing mix and Internet consumers' decision-making styles, we develop an ontology for e-marketing mix that is mapped with Internet Consumers' decision-making styles. This ontology defines the concepts for the relationships between the E-Marketing mix model and the psychological factors of Internet consumers. Based on the understanding of Internet consumers' decision-making styles, businesses can make use of their corresponding e-marketing mix to match with individual Internet consumers' decision-making styles in order to attract more targeted Internet consumers. As a result, it can generate profitable and sustainable revenue streams.**

I. Introduction

Today there are billions of home pages on the web; many of which provide an E-commerce environment. From the consumers' viewpoint, they do not know which web sites are suitable for them to buy their products and need to choose a business web site that matches their preferences. They may be confused by the number of business web sites available to them. From the business service providers viewpoint, they cannot increase their revenue if their target consumers are confused by the plethora of business web sites that are available to them.

For different types of products, Sam K. M. & Chatwin C. R. [1] introduced different consumers' decision-making styles in an E-Commerce environment. Based on those decision-making styles, Sam K. M. & Chatwin C. R. [2] established the relationship between businesses' E-Marketing Mix and Internet Consumers' decision-making styles in E-Commerce. This relationship is important to describe as we can create a situation where both of the two-parties become winners in the E-Commerce environment. In this paper, this relationship is defined by developing an ontology, which is described as machine interpretable definitions of basic concepts in the domain and the relationships between them [3].

Tools used to develop the ontology: Protégé 2000 and Jess

1) What is Protégé?

Protégé is a tool that can allow users to create a model and collect information. It is a rich modelling language that can show inheritance relationships, constraint "overriding" and expresses "webs" of relationships.

2) Why use Protégé?
   A. The ontology editior is free and open source.
   B. Protégé ontologies can be exported into a variety of formats including RDF(S), OWL and the XML Schema.
   C. Protégé is based on Java, which is extensible and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development.

3) What is Jess (Java Expert System Shell)?

It is a free license rule-based engine software developed at Sandia National Laboratory for non-profit organizations.

4) Components of Jess Inference (Rule) Engine
   A. **Pattern Matcher** – decides what rules to fire and when.
   B. **Agenda** – schedules the order in which activated rules will fire.
   C. **Execution Engine** – is responsible for firing rules and executing other code.

5) Why use Jess?
   A. It is easy to create rules in Jess
   B. Jess can be integrated with Protégé. Fig. 1 shows the relationship between Protégé and Jess.
   C. Jess is also integrated with Java as it is a fully developed Java API. As a result, it is easy to create rule-based expert systems.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 279–282.

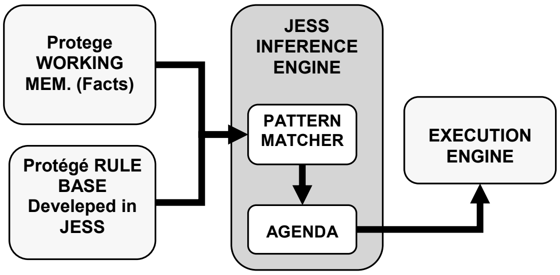


Fig. 1. Relationship between Protégé and Jess

II. OUR APPROACH

Since businesses' E-marketing mix and consumers' decision-making styles are characterized by two different models, we use rules to integrate the two models together. Before describing the rules, the design of classes for these two models are described as follows:

Businesses' E-Marketing Mix Model represented in the Ontology

All instances of the class Model have a slot SoldByComp, the value of which is an instance of the class Comp (Fig. 2). All instances of the class Comp have a slot Model_Selling, the value of which is an instance of the class Model. All instances of the class Model have a slot BelongTo, the value of which is an instance of class Product.

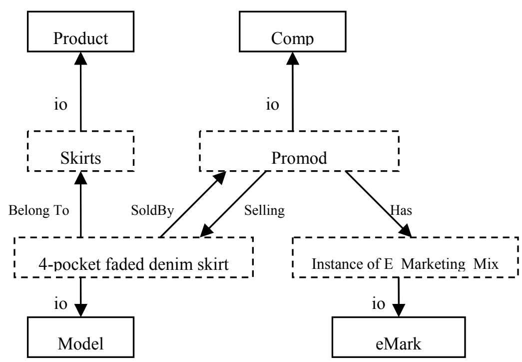


Fig. 2. Some classes, instances and relationships between them in the business domain. Black box: class and dotted box: instance. Links represent slots and instance of.

The following figure (Fig. 3) shows the slots of the class Comp (Company). In a business web site, it offers many different product models. Therefore, the cardinality of the slot Model_Selling is multiple. Furthermore, each business web site has its own unique E-Marketing Mix, which is a slot in this class.

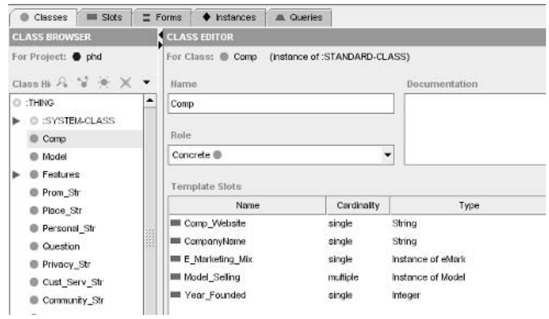


Fig. 3. Snapshot from the slots of the class Comp

Inside the eMark class, it defines the different elements of the e-marketing mix model as shown in Fig. 4. Each element, in turn, has different elements supporting their corresponding e-marketing mix elements as shown in Fig. 5.

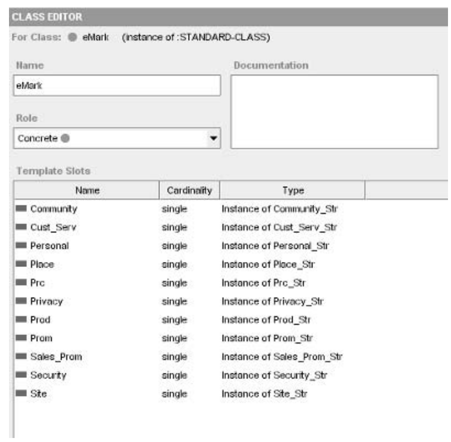


Fig. 4. Different Elements of E-Marketing Mix indicated by slots in eMark class

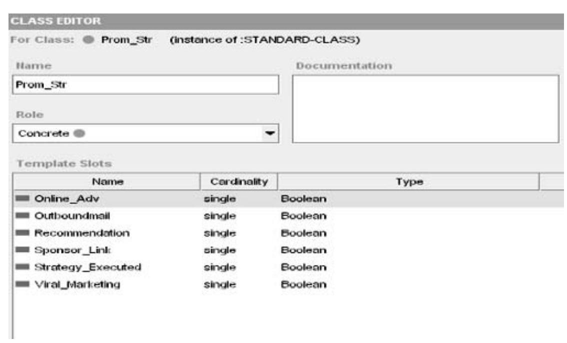


Fig 5. Different supporting elements for Promotion Strategy

Internet Consumers' Decision-Making Styles represented in the Ontology

All instances of each person class have different psychological factors, including price factor, etc. All instances of the Price Conscious factor have slots Price_Q1, Price_Q2, Price_Q3 (corresponding to three questions in price conscious factor), and Factor_Average (average score of the corresponding factor) as shown in Fig. 6.

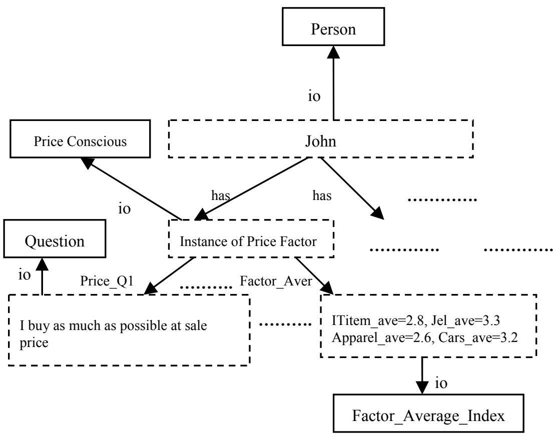


Fig. 6. Relationship among classes in consumers' psychological factors domain. Black box: classes and dotted box: instances. Links represent slots and instance of.

In Fig. 7, there is a class for each psychological factor, which is important in describing the concepts of this model so that it is easier to access for each psychological factor.

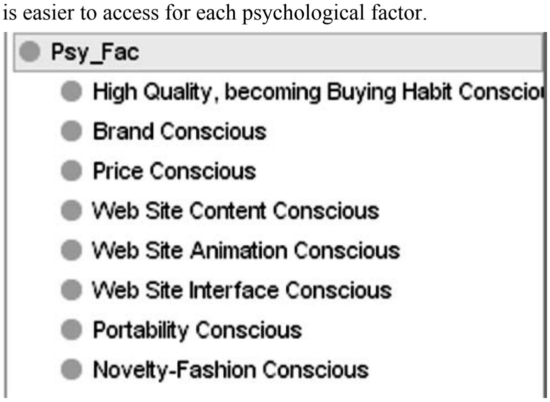


Fig. 7. Hierarchical Structure of Psychological Factor class.

Issues in E-Marketing Mix Model

There are some questions that need to be solved inside the E-Marketing Mix Model:

1. How to determine whether each marketing mix element is being executed or not in the E-Marketing Mix Model?
2. How to determine the value (True or False) for each marketing tool supporting each marketing mix element?

With regard to Question 1 above, the number of marketing tools that have been provided for a particular strategy by the business company should be counted and divided by the total number of marketing tools for that particular strategy. The percentage is then used to determine how much of the corresponding strategy is executed, assuming that the weights of all supporting tools for that particular strategy are equal. The marketing mix element ratio will be used to find out how suitable the web site is for the consumers. Only if the ratio is ≥ 0.5, we can say that the strategy is executed by the business web site.

As for Question 2 above, for each marketing tool, there are some related keywords which are being used to find out whether a particular marketing tool has been adopted by the business web site. Fig. 8 shows the keywords associated with a particular marketing tool.

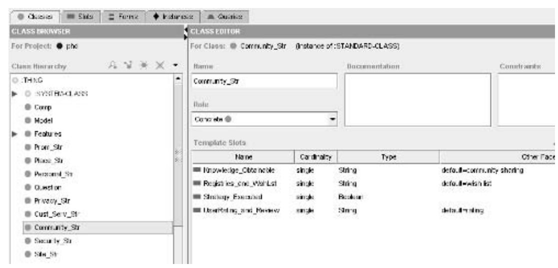


Fig. 8. A keyword associated to each supporting e-marketing tool in Community Strategy.

Each associated keyword of the corresponding marketing tool is used to search each web site to find out whether the business web site has been adopting the marketing tool for the e-marketing mix element.

If the keyword is found, the corresponding marketing tool has been adopted.

Issues in Consumers' Psychological Factors Model

There is a set of rules inside Consumers' Psychological Factors Model:

1. How to determine the score of each psychological factor for each consumer?
2. How to determine whether the score is average or not?

For Question 1, the score of each psychological factor is determined by adding the scores for each related question and then dividing by the total number of related questions.

For Question 2, suppose a consumer wants to buy an ITitem, If (Person.Price_Factor.Score >= Person.Price_Factor.Factor_Average.ITitem_Average_Index) then the score is on average

Relationship between businesses' E-Marketing Mix Model and Consumers' Psychological Factors

In order to establish the relationship between businesses' E-Marketing Mix Model and Consumers' Psychological Factors, a set of rules need to be developed as below and the rules should match each of the business web sites which offer suitable products to consumers:

State_1:
```
Confidence_Level = 10
If   (Person.High_QLY_Buy_BHT.On_Average)
        If (Comp.E_Marketing_Mix.Personal.Strategy_Executed &
        Comp.E_Marketing_Mix.Community.Strategy_Executed)
                Go to State_2
        Else
                Confidence_Level = Confidence_Level – 1
                Go to State_2
Else
            Go to State_2
```

State_2:
```
If (Person.Brand_Factor.On_Average)
        If (Comp.E_Marketing_Mix.Personal.Strategy_Executed)
                Go to State_3
        Else
                Confidence_Level = Confidence_Level – 1
                Go to State_3
Else
        Go to State_3
```

State_3:
```
If (New_Style.On_Average)
        If (Comp.E_Marketing_Mix.Prod.Strategy_Executed &
            Comp.E_Marketing_Mix.Prom.Strategy_Executed)
                Go to State_4
        Else
                Confidence_Level = Confidence_Level – 1
                Go to State_4
Else
        Go to State_4
```

State_4:
```
If (Price.On_Average)
        If (Comp.E_Marketing_Mix.Prc.Strategy_Executed &
            Comp.E_Marketing_Mix.Prom.Strategy_Executed &
            Comp.E_Marketing_Mix.Sales_Prom.Strategy_Executed)
                Go to State_5
        Else
                Confidence_Level = Confidence_Level – 1
                Go to State_5
Else
        Go to State_5
```

State_5:
```
If (Web_Site_Content.On_Average = False)
        Go to State_6
Else
        If (Comp.E_Marketing_Mix.Cust_Serv.Strategy_Executed &
        Comp.E_Marketing_Mix.Community.Strategy_Executed &
            Comp.E_Marketing_Mix.Site.Strategy_Executed &
            Comp.E_Marketing_Mix.Security.Strategy_Executed)
                Skip
        Else
                Confidence_Level = Confidence_Level – 1
                Go to State_6
```

State_6:
```
If (Web_Animation.On_Average or Web_Interface.On_Average)
        If (Not Comp.E_Marketing_Mix.Site.Strategy_Executed)
                Confidence_Level = Confidence_Level – 1
```

Finally, the consumers can choose their preferred business web sites based on their confidence level.

III. Conclusion

Different internet consumers have different psychological factors and different E-business companies have different E-marketing mix models. Based on the ontology and the rules mapping psychological factors and E-marketing mix model, further enhancement of the satisfaction-of-needs of both the internet consumers and E-business companies has been achieved.

REFERENCES

[1] Sam K. M. & Chatwin C. R., "Multi-product Generalizability of a Scale for Profiling International Internet Consumers' Decision Making styles in E-Commerce," The Conference on Information Management in Modern Enterprise Proceedings, p.132-138, 2005

[2] Sam K. M. & Chatwin C. R., "The Mapping Between Business E-Marketing Mix And Internet Consumers' Decision-Making Styles In E-Commerce," The Fifth International Conference on Electronic Business, 2005

[3] Natalya F. Noy and Deborah L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology

Monitoring Choreographed Services

L. Ardissono and R. Furnari and A. Goy and G. Petrone and M. Segnan
Dipartimento di Informatica, Università di Torino
Corso Svizzera 185, 10149 Torino, Italy

***Abstract.* Web Service choreography management challenges the research on Service Oriented Architectures: different from Web Service composition, where a central orchestrator process invokes the Web Service suppliers, a choreographed service is decentralized and it is based on the coordination of a set of Web Service suppliers having a partial view of the overall service. In this paper, we present a framework which supports the monitoring of the progress of a choreographed service, the early detection of faults and the notification of the Web Services affected by the faults.**

I. Introduction

In the research on Service Oriented Architectures [9], the management of supply chains has been based on processes which orchestrate the execution of services within possibly complex workflows. For instance, the WS-BPEL language [7] is used to specify parameters and execution order of the invocations of operations on Web Service suppliers, as well as of the internal activities to be performed by the composite Web Service. Moreover, Abstract Processes, based on the WS-BPEL syntax, describe the invocation protocol of a conversationally complex Web Service supplier; this specification enables consumers to invoke the operations without violating the suppliers' business logics.

Web Service orchestration has been introduced to manage hierarchical supply chains in which a centralized entity invokes the suppliers. However, it has been recognized that, even in Enterprise Application Integration scenarios, the management of a non trivial supply chain often imposes the decentralized management of the composite service [5], e.g., in order to deal with the business requirements of suppliers which serve multiple consumers. In such situations, the cooperating Web Services have partial views of the overall service and they have to coordinate with each other in a complex multi-party conversation denoted as *choreography* [10]. The successful execution of the choreographed service thus depends on at least two factors: when the service is designed, Web Service suppliers have to be bound to each other (possibly by employing mediation services for interoperability purposes [6]); at service execution time, they have to perform operations successfully and in due time.

In this paper we propose a framework which supports the monitoring of the progress of a choreographed service, the early detection of faults and the notification of the Web Services affected by the faults. In our framework, a Monitor Web Service tracks the execution of the cooperating Web Services by analyzing their conversational behavior. During the execution of the choreographed service, the Monitor is informed about the messages sent or received by the cooperating Web Services and about their execution state. The Monitor utilizes this information to check whether the overall service correctly progresses, i.e., if the message flow among the Web Services is consistent with the choreography. If any fault occurs, the Monitor evaluates whether the choreographed service can still complete and informs the Web Services, in order to let them react to the occurred problem. For generality purposes, we assume that the Monitor does not have any information about the internal implementation of the Web Services. Thus, our framework relies on the analysis of messages.

Our choreography framework builds on WS-Coordination [2] to manage the coordination context between the Web Services but it replaces the Web Services Coordinator with a Web Services Monitor which proactively checks the progress of the choreographed service and propagates the coordination information. In the rest of this paper, Section 2 shows the benefits of monitoring on a sample scenario, Section 3 presents our framework, and Section 4 discusses some related work and concludes the paper.

II. Motivating Example

Monitoring the progress of a choreographed service and notifying the cooperating Web Services about execution problems enables to avoid delays in execution and to react to faults in timely fashion. In fact, the data dependency chains among different Web Services may be long and a Web Service might become aware that it is affected by an execution problem occurred in another Web Service after several messages have been sent. We show this point on a simple example from a Business to Consumer (B2C) e-commerce scenario.

In a typical B2C service, the online store is the contact point with the end customer; given the customer's orders, the store invokes a warehouse to prepare the requested parcel. In turn, the warehouse service might inquire with some shippers about the cost of sending the parcel to the

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 283–288.

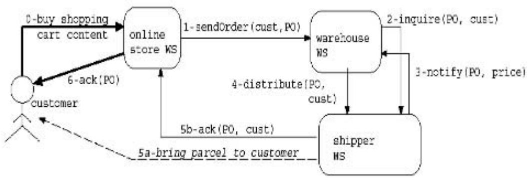


Fig. 1. Collaboration Diagram of a choreographed service.

customer in order to choose the most convenient one. Thus, the online store would not directly control the shipper's activities. However, in order to be able to inform the end customer about possible delivery problems (e.g., delays), the online store needs an acknowledgement from the selected shipper service as soon as the parcels have been sent out. Figure 1 shows the messages exchanged by the cooperating Web Services in the described scenario.

Now, suppose that, after having received an order from the warehouse, the selected shipper service becomes unavailable because of a software problem. Then, the online store would wait for the shipper's acknowledgment until a possibly long time out expires, and it would not be able to inform the end customer about the delay in a timely fashion. By monitoring the situation, the time out might be anticipated and the online store might be informed about the problem as soon as possible.

This example is deliberately simple but it should be noticed that, in a realistic scenario, the number of cooperating Web Services might be large and the dependency graph very complex. Thus, several service consumers might be blocked by the failure of a supplier. Having a global view of the situation is therefore strategic to notify the services which cannot continue their own execution in due time.

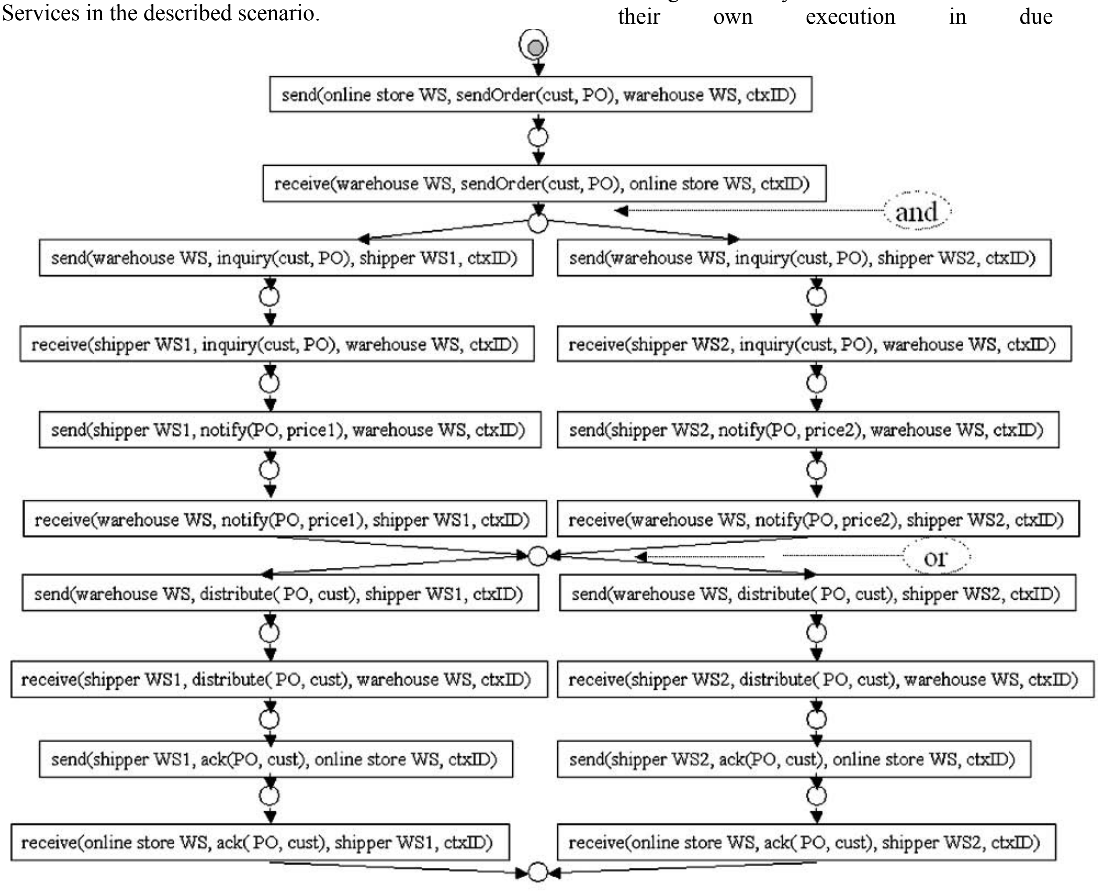


Fig. 2. Choreography specification of the online store example.

III. Our Framework

A. Choreography Specification

The choreography of a complex service specifies the admissible message exchanges which may occur between all the cooperating Web Services at execution time.

It should be noticed that alternative service providers might be employed during different executions of the same choreographed service, because of availability problems or of different evolutions of the overall service. Therefore, it is convenient to generate the choreography specification at service execution time, instead of pre-compiling a large choreography which would include all the possible alternatives. For this reason, we have adopted an incremental approach to the specification of the choreography of a complex service. Our approach is based on the definition of *local views* on the choreography, held by the cooperating Web Services, and on the composition of such local views at service invocation time. The basic assumption is that each Web Service provider must publish the specification of its own interaction protocol (as suggested in Web Service conversation standards, e.g., WS-CDL [12] and WS-BPEL Abstract Processes), which might be manually defined by its own developer.

Let's suppose that the interaction protocol of each cooperating Web Service WS_i is represented as an Abstract Process. Then, the local view of WS_i on the choreography is determined by binding its Abstract Process to the endpoint Web Services interacting with WS_i. Moreover, the specification of the overall choreography may be achieved by composing the local views of all the cooperating Web Services.

We assume that, at choreographed service design time, the candidate Web Service providers to be invoked are selected by taking their Abstract Processes into account, and that the developer of the service solves any possible conflict due to heterogeneous interaction protocols (e.g., by adding mediators to the pool of cooperating Web Services [6]). Under these circumstances, the local views may be composed incrementally.

We have adopted a workflow language for the representation of the choreography because this is a possibly complex process; the only difference with standard workflows is the fact that the performed activities are conversational actions. For instance, Web Services may send messages sequentially, in parallel, or they may follow alternative message flow paths, which may split and join in the various phases of execution of the choreographed service. Specifically, we adopt a Reconfigurable Petri Net representation [3], which can be utilized to check whether the cooperating Web Services

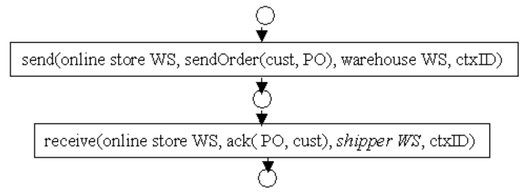


Fig. 3. Online store WS's local view on the choreography.

send messages as expected, and to represent the progress of the choreographed service by following the message flow paths corresponding to the sent/received messages. This type of Petri Net can be used to model the condition in which several choreographies are merged and thus the network topology changes during the service execution.

Figure 2 depicts the complete choreography specification of our sample scenario. Notice that the activities of the choreography have a *ctxID* parameter, which identifies the choreography instance to be considered by the Monitor. This is necessary because, at each time, the choreographed service might have multiple running instances; moreover, the Monitor might serve different choreographed services at the same time.

- Transitions represent the act of sending or receiving a message. A *send* transition specifies the sender, the object message (i.e., the message which has been sent), the recipient of the message and the choreography context identifier *ctxID*. The *receive* transitions are symmetric (they represent the conversational action performed by recipients) and specify recipient, object message, sender and choreography context ID.
- States represent the evolution of the choreographed service while messages are being sent or received.

Figures 3 and 4 depict the local views on the choreography held by the online store and the warehouse Web Services. In the figure, the roles not yet instantiated on concrete Web Service endpoints are shown in italics. For instance, initially, the online store WS does not know whether the shipper WS1 or WS2 will distribute the goods to the customer; thus, the *shipper WS* parameter is reported in italics.

B. Initialization of the Choreography Monitoring Process

Similar to the approach adopted in WS-Coordination, the first Web Service participating to a choreographed service invokes the Monitor, which generates a choreography coordination context and starts monitoring the messages between Web Services. Then, each invoked Web Service registers to the Monitor in order to join the pool of participants. As explained in Section III.D, when a

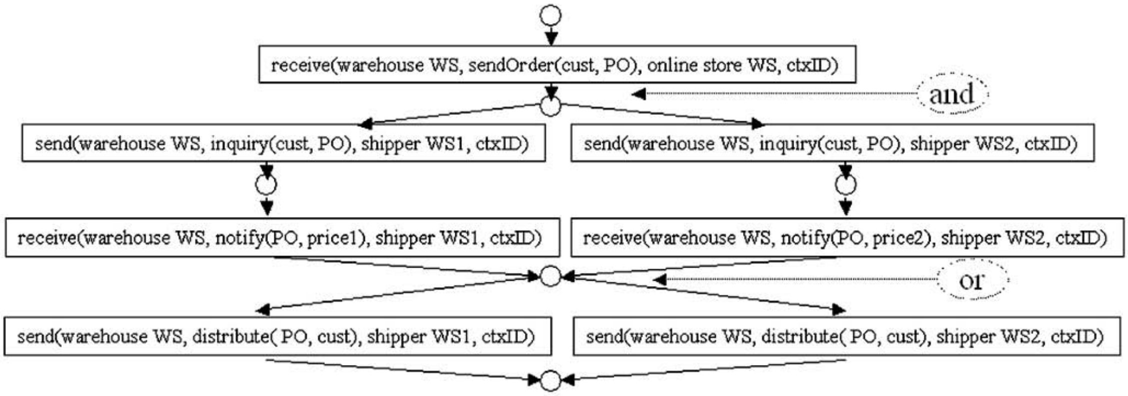


Fig. 4. Warehouse WS's local view on the choreography.

cooperating Web Service registers, it also provides the Monitor with its local view on the choreography. Thus, the Monitor starts with an empty choreography and incrementally extends it during the execution of the choreographed service.

As the choreography specification can be completed only after all the cooperating Web Services have registered, at each stage of execution, the Monitor can only detect interaction problems on the subset of the whole choreography built so far. However, this is sufficient for the early detection of several problems such as the identification of execution paths which cannot be successfully terminated because of an occurred fault.

C. Choreography Monitoring

During the execution of the choreographed service, the Monitor utilizes the partial service choreography to track the progress of the registered Web Services, and to identify possible execution problems which might obstacle their completion. Intuitively, while Web Services interact with one another, the Monitor follows their conversational behavior on the choreography net. The configuration of tokens in such a Petri Net describes the progress state of the choreographed service, from the conversational point of view. Specifically:

- The Monitor starts the choreography simulation from the initial state of the net.
- Each time a Web Service sends or receives a message, it notifies the Monitor about the event (see Section III.D). In turn, the Monitor performs a transition in the choreography net, moving tokens to represent the fact that a conversational action has been performed. As the choreography may include parallel paths, multiple nodes in the graph can be active at the same time. The active nodes represent the conversational state after several Web Services have sent or received messages.

If the cooperating Web Services successfully terminate the execution, the choreographed service completes and the Monitor reaches the final state of the choreography. Conversely, if a failure occurs, the Monitor analyzes the possible continuations of the interaction between the Web Services in order to identify which ones are affected by the problem. Thanks to the Petri Net representation, the Monitor can adopt existing dead path elimination algorithms to reason about the possible continuations of the interaction. For instance, given the current state of the choreography net and the information about the execution state of the Web Services (e.g., some is faulted), the Monitor can decide whether it is still possible to complete the choreographed service, and which Web Services should terminate their execution because they are involved in a dead path. Moreover, the Monitor may notify other affected Web Services about possible execution problems.

Let's consider our sample choreography and suppose that the shipper WS2 fails immediately after the warehouse WS has sent the two *inquiry(cust, PO)* messages. Then, the choreographed service may successfully continue if the warehouse WS invokes WS1 for the delivery. Nevertheless, the warehouse WS should be informed about the failure in order to avoid waiting for an answer from the shipper WS2, until the expiration of a time out. As another example, if the shipper WS1 fails after having received the *distribute(PO, cust)* message from the warehouse WS, it would be useful to inform the online store WS about the failure. In fact, the online store WS might immediately notify the end customer that the delivery of goods is going to be delayed.

D. Interaction between Cooperating Web Services and Monitor WS

In order to be monitored and to be informed about execution problems concerning the choreographed service, the cooperating Web Services should offer a set of WSDL operations to be invoked by the Monitor WS; moreover, they should invoke some operations on the Monitor to inform it about the messages they send and/or receive. Specifically, the interaction between Monitor WS and cooperating Web Services is organized as follows:

- When an instance of the choreographed service is started, the first Web Service in execution sends the Monitor WS a *startConversation(originatorWS)* message.
- Upon receiving this message, a new instance of the Monitor WS is created. That instance creates a new choreography coordination context, registers the Web Service and returns a context ID by sending the Web Service a *receive(ctxID)* message. The Web Service also sends the Monitor its local view on the choreography (*addChoreography* message).
- When a registered Web Service invokes a Web Service provider, it adds the choreography coordination context ID (*ctxID*) to its own message. At the first invocation, the Web Service provider registers to the Monitor WS by specifying the *ctxID* (*register(senderWS, ctxID)*).
- Immediately after registration, each cooperating Web Service sends the Monitor WS a request to add its local view of the choreography to the partial choreography by sending a message of type *addChoreography(senderWS, localView, ctxID)*.
- Each registered Web Service receives choreography coordination messages from the Monitor WS. Moreover, the Web Service can send messages to inform the Monitor about changes in the execution state. For instance, the Web Service might notify the Monitor about an occurred failure or about its successful termination.
- The Monitor WS can proactively send *getStatus* messages to the cooperating Web Services, which respond with *status* messages describing their own execution state. The *getStatus* message can be seen as a sort of *ping*: if the invoked Web Service does not respond in due time, this means that it is unavailable.[1]

[1] A missing reply might also mean that the network connection between Web Service and Monitor is down. In order to take this issue into account, we will extend the interaction protocol to support information redundancy.

We assume that the registered Web Services may send the Monitor various types of messages in order to inform it about their activities. To this purpose, the Monitor WS offers the following WSDL operations:

- *send(senderWS, msg, destWS, ctxID)* informs the Monitor that the sender WS has sent message *msg* to the destination WS, within the instance of the choreography identified by *ctxID*.
- *receive(destWS, msg, senderWS, ctxID)* informs the Monitor that message *msg* has been received by the recipient Web Service.
- *startConversation(originatorWS)*: the Web Service starting the choreographed service notifies the Monitor that the process has started and has to be monitored.
- *addChoreography(senderWS, localView, ctxID)*: the *senderWS* Web Service sends the Monitor its own local view of the choreography, by specifying the choreography instance to be considered (*ctxID*).
- *register(senderWS, ctxID)* notifies the Monitor that *senderWS* joins the service.
- *status(ctxID, currentStatus)*: upon receiving a *getStatus* message from the Monitor, the Web Service sends its current execution status.

In order to test our framework, we have developed a *WS-monitoring Java class*, which implements the WSDL operations to be offered by a cooperating Web Service, and can be exploited by a Web Service supplier to interact with the Monitor WS without modifying its own business logic; e.g., the class may be used within message handlers associated with each cooperating Web Service. Moreover, we have developed a Monitor WS which runs a choreography net in order to track the progress of a choreographed service. The Monitor is based on the JBPM workflow management tool and the choreography net is implemented as a JBPM process.

IV. DISCUSSION

We have described a monitoring framework supporting the early detection of faults in choreographed services, i.e., complex services characterized by the cooperation of multiple service providers, none of which has complete control on the overall business logic. Our framework is based on the introduction of a Monitor Web Service, which maintains a snapshot of the multi-party conversations carried out by the cooperating Web Services. When a failure occurs in an instance of the choreographed service, the Monitor analyzes the choreography specification to decide whether it is still possible to continue the service. Moreover, the Monitor notifies the service providers which cannot continue their execution, in order to let them react as needed.

Our monitoring framework builds on the WS-Coordination standard [2] for the management of the coordination context between a set of cooperating Web Services. However, our Monitor WS offers additional features with respect to a standard Coordinator; e.g., it composes the local views on the choreography held by the Web Services into a unified one; moreover, it analyzes the evolution of the choreography in order to assess its progress state and notify the cooperating Web Services about execution problems.

Indeed, there are similarities between our approach and the one followed in WSBusinessActivity, and some of the messages exchanged between cooperating Web Services and Monitor WS are the same as those defined in that standard. However:

- We introduce new message types in order to inform the Monitor about which choreography paths are traversed during the execution of the overall service.
- The Monitor utilizes the choreography specification to evaluate the possibility of success of the overall service, depending on the execution state of the cooperating Web Services and on which portion of the choreography has been completed. The reason is that, while the failure of a business activity determines the failure of the whole scope to which the activity belongs, the occurrence of a failure during the execution of a choreography does not necessarily cause the failure of the whole service. In fact, alternative courses of actions may exist, which enable the service to successfully complete.

Our framework could be extended to provide additional monitoring features. For instance, it would be interesting to deal with delays in service execution in order to address some typical Quality of Service concerns. To this purpose, the Monitor WS should be extended with a temporal reasoning module which estimates the time needed to complete a choreographed service. If timing information were added to the specification of a choreography (e.g., time outs and expected time to elapse between messages), the Monitor might compute the expected service execution time and check the progress of the service. This would enable the Monitor to evaluate whether it is still possible to complete the service on time, or if any admissible conversation paths have to be excluded because they would violate the overall time constraints.

Adding a Monitor to a choreographed service is interesting only when the interaction between the cooperating Web Services is complex and simply restarting the whole process is costly. In fact, monitoring imposes an obvious overhead due to an increase in messages exchanged between the Web Services and the Monitor. Moreover, monitoring the progress of a choreographed service raises serious applicability issues; on the one hand, the introduction of a centralized observer controlling the overall message flow is knowledge intensive, unless the choreography specification can be automatically generated out of the local views of the cooperating Web Services. On the other hand, various languages are being proposed to specify the interaction protocol of a Web Service (e.g., BPEL Abstract Processes [7] and automata-based approaches such as the Abstract State Machines [1]) and the choreography of a set of cooperating Web Services (e.g., WSCDL [12] and Petri Nets [11]). However, an agreement on representation formalisms has not been reached so far; e.g., see [4]. Therefore, the choreographed service might need to abstract from heterogeneous representations of the interaction protocols of the individual service providers.

ACKNOWLEDGMENT

The work described in this paper has been partially funded by the EU Project WS-DIAMOND (grant IST-516933) and the MIUR Project QuaDRAnTIS.

REFERENCES

[1] M. Altenhofen, E. Börger, and J. Lemcke. An abstract model for process mediation. In *Proc. 7th Int. Conf. on Formal Engineering Methods*, pp. 81–95, Manchester, UK, 2005.

[2] Arjuna. Introducing WS-Coordination - an Arjuna Technologies report. http://www.arjuna.com/library/reports/2003-04-WS-Coordination-intro.pdf, 2003.

[3] E. Badouel and J. Oliver. Reconfigurable Nets, a class of high level Petri Nets supporting dynamic changes. Technical Report PI1163, Inria, 1998.

[4] A. Barros, M. Dumas, and P. Oaks. A critical overview of the Web Services Choreography Description Language (WS-CDL). *BPTrends*, March, 2005.

[5] G. Chafle, S. Chandra, V. Mann, and M.G. Nanda. Decentralized orchestration of composite Web Services. In *Proc. of 13th Int. World Wide Web Conferenc*, pp. 134–143, New York, 2004.

[6] DERI International. Web ServiceModelling eXecution environment. http://www.wsmx.org/, 2006.

[7] OASIS. OASIS Web Services Business Process Execution Language. http://www.oasisopen.org/committees/documents.php?wg abbrev=wsbpel, 2005.

[8] OASIS. OASIS Web Services Business Activity Framework (WS-BusinessActivity). http://schemas.xmlsoap.org/ws/2004/10/wsba/, 2006.

[9] M.P. Papazoglou and D. Georgakopoulos. Service-Oriented Computing. *Communications of the ACM*, 46(10), 2003.

[10] C. Peltz. Web Services orchestration and choreography. *Innovative Technology for Computing Professionals*, 36(10):46–52, 2003.

[11] W. van der Aalst, M. Dumas, C. Ouyang, A. Rozinat, and H.M.W. Verbeek. Choreography conformance checking: an approach based on BPEL and Petri Nets. In *ePrints Archive*, Adelaide, South Australia, 2006.

[12] W3C. Web Services Choreography Description Language version 1.0. http://www.w3.org/TR/ws-cdl-10/, 2005.

Version Manager: A step towards Synthetic-Warehouse-Builder extension

M. Khurram Shahzad
khur-sha@dsv.su.se
Department of Computer Science,
M.A Jinnah Campus, Defense Road, off Raiwand Road,
COMSATS Institute of Information Technology, Lahore, Pakistan.

Abstract--To achieve vital advantages of simulating business scenarios, extensions are made to versioning concept by dividing alternative versions into two types, Materialized Alternative Versions and Virtual Alternative Versions. This paper presents querying method from multi-version data warehouse in the presence of real versions, materialized and virtual alternative versions. This is done by making extensions to our Synthetic Warehouse Builder with new component called Version Manager.

Version manger with its confinement process divide query into smaller parts called mono-version queries; execute it on relevant and heterogeneous version/s and merging the result to present it for analysis. It is expected that, Version Manager is very useful for short term data requirements as well as rapidly changing requirements of simulating business alternatives.

Keywords-- Data Warehouse, evolution, versioning, querying DW, architecture, data retrieval.

I. INTRODUCTION

Data Warehouse (DW) provides integrated access to transactional data source/s. But dynamics of this source/s has resulted in derivation of inconsistent analytical results from DW [1]. These dynamics can be of two types: a) Content changes, due to execution of data manipulation language b) Schema changes, due to execution of data definition language [2].

Various naive approaches have been proposed to handle business dynamics, these approaches can be categorized, schema evolution [3], Versioning approach [4]. With evolution approach, schema is upgraded, but has high maintenance cost [5].

For better what-if-analysis and to improve the quality of decision-making the concept of simulating business scenarios has been proposed [1,6], but conventional DW does not support simulating business scenarios. Hence, two types of versions are proposed: a) real version b) alternative versions. The DW which maintains these versions is called multi-version data warehouse (MV-DW). Retrieval from MV-DW has been made possible by multi-version query language [7].

For solving querying problems of we have developed Synthetic Warehouse Builder (SWB) [8]. SWB provides transparent access to users for querying multiple versions of DW without caring about implementation details of versions. Recent study shows that vital advantages of simulating business scenarios can only be achieved by using three types of versions a) Real versions b) Materialized Alternative Versions c) Virtual Alternative Version [9].

Observations-- a) SWB do not provide flexibility of simulating business scenarios in MV-DW b) Vital advantages of, simulating business scenarios, cannot be achieved by exploiting alternative versions c) The conception of handling simulating business scenarios by alternative versions is required to be re-worked d) To extend the functionality of SWB, Version-Manager is required.

Contribution-- The objective of this paper is 1) to present the architecture of Version-Manager (an added component of SWB) to provide flexible environment for what-if-analysis and simulating business scenarios in order to improve the quality of decision-making. 2) Querying method for three types of versions, without letting the users to know about location of data in any version.

Rest of paper is organized as follows; section no 2 gives the motivation for extending the SWB functionality, section 3 overviews existing approaches to address evolutionary problems of DW and simulating business alternatives. Introduction to SWB, its properties along with functionalities are given in section 4 while section 5 presents the proposed architecture, and query processing using Version Manager is given in section 6 of the paper. Finally, paper is concluded in section 7 with small discussion.

II. MOTIVATION

In today's knowledge oriented society, success of organization depends upon quality of decision-making [5]. Simulating business scenarios for better what-if analysis has recently been proposed [6, 1] by maintaining real and alternative versions, but various businesses are found to have vibrant simulating scenarios i.e. they can be changed at runtime. Since alternative versions have instances attached with them so changes are

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 289–293.

absolutely not possible to be made, hence restricts the scope of the user's what-if-analysis.

For achieving vital advantages of alternative business scenarios and providing users with a flexible simulating environment, alternative versions are divided into two types [9].

a) Materialized Alternative Versions (MAV), have instances attached with them and can also share data with its child versions. MAV's are used for simulations, whose structure are static and are used most of the time. These are also called static simulations.
b) Virtual Alternative Versions (VAV), have no instance attached with them instead they are stored like virtual tables in multi-version catalog and gets data on-the-fly from mapped set of real and materialized alternative versions. These types of alternative versions are used for vibrant (dynamic) simulations.

Storing and mapping of versions is not the real task instead it is required to retrieve data from various versions and make it useful for malicious decision-making, which has not been done.

SWB provides transparent access to real and alternative versions, but does not provide flexible simulating environment, hence raises the need for addition of component called Version-Manager (VM). VM is expected to not only provided interface for simulating business scenarios and run-time addition of virtual dimension and fact tables but also provide facility for querying a) Real Versions b) Materialized Alternative Versions c) Virtual Alternative Versions. The only extra effort required is one-time mapping of VAV's to respective MAV/s and RV/s.

III. RELATED WORKS

Transactional systems act as a source of data for DW. It is found that changes to these sources may result in derivation of inconsistent results [1, 5]. Semi-Star [5] can handle these changes but are suitable only for small and medium sized dynamic organization. Other methods proposed to handle these changes are: a) Schema and data evolution method [3, 4], according to this approach changes to schema are applied to new schema and data is transported to new schema. But this approach gives unsatisfactory response time in the presence of increased number of changes [8]. b) Temporal versioning method [10, 11], in which changes new DW versions are created with a time stamp but its disadvantages has already been discussed in [12,13] and quoted in [1].

For achieving the objective of better what-if-analysis, concept of simulating business alternatives has been reported [1], which can be handled along with source dynamics by using two types of versions i.e. real and alternative versions. For querying purpose multi-version query language has been proposed in [7] by extensions to standard query language. We have already removed shortcomings by building SWB for transparent querying method [8].

But, a) Vital advantages of simulating alternative business scenarios (SBS) has not been achieved b) It is required to provide users with a flexible environment for using simulations to improve quality of decision-making c) SWB has restricted querying scope i.e. only the set of three querying possibilities exists, that are: i) Querying only current DW version ii) Query the set of real DW versions iii) Or querying set of selected alternative DW versions. These issues can only be addressed by adding Version Manager to SWB.

IV. SYNTHETIC WAREHOUSE BUILDER [8]

SWB provides transparent access to set of real and alternative versions. While using SWB users are not required to keep track of set of versions, their derivation hierarchy as well as instances attached with the versions, instead only the knowledge of logically integrated schema will work.

Transparency in SWB has been defined at three levels [8] a) Source transparency, data stored in different versions are made transparent to users. i.e. user will retrieve data from multiple versions independent of the existence of source table/s in version/s, but it should be present in at least one of the versions.

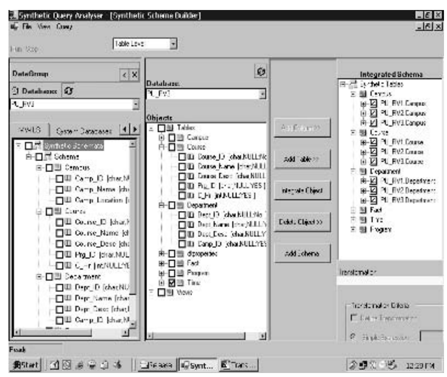


Fig.1. MV-ILS Builder [8]

b) Projection transparency, user should write the query independent of existence of attribute in one or more tables of different versions.
c) Analysis transparency, in the presence of this type of transparency, user will write query independent of predicate-attribute present in one or more versions. But it should be available in at least one version. The advantage of fully transparent access to DW is high level support, which it provides for development of complex relational multi-version DW.

Three level of transparency is provided with the help of two major components: a) MV-ILS builder, acts as a core of providing integrated access to the users by provides logically integrated schema that contains the union of all the participating set of versions [8], shown in fig. 1.

b) Synthetic Query Analyzer (S-analyzer), shown in fig. 2 is produced to handle querying issues of real and alternative versions of DW. Input query to it is called multi-version query or transparent query (executable on multiple-versions), which is further transformed to mono-version queries by confinement process.

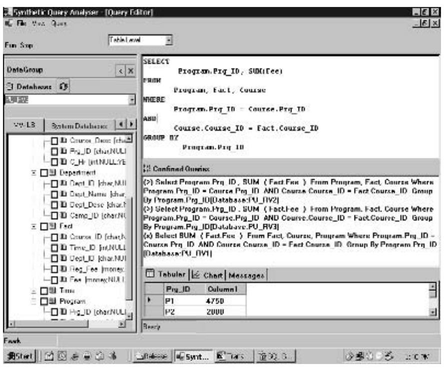


Fig.2. Synthetic Query Analyzer [8]

Although, SWB facilitate users by providing transparent access to versions but do not provide flexible environment for simulating business scenarios. So, need for extensions to SWB is proposed in [9] which can be met by adding small component named Version-Manager with the abilities of addressing issues related to three types of versions.

V. VERSION-MANAGER

Version manager extends the functionality of transparent querying from multiple real and alternative versions to: a) Set of real, materialized and virtual alternative versions b) Flexible environment for simulating business scenarios.

VAVs have no instance of its own, instead selected real and alternative versions acts as a data source for such versions [9]. All this is done on-the-fly, although some of the performance issues are evolved but it can be handle by using the strengths of distributed and grid computing [14]. Moreover, the performance can be maintained by using appropriate set of materialized and alternative versions.

Architecture of Version Manager is shown in fig. 3. (a), it provides the facility of breaking a query say TSQL, into possible set of mono-version queries i.e. RV-SQL, AV-SQL and VSQL. Each query will then be executed on relevant version/s. Since virtual alternative versions have no instance attached with them and they are mapped to real and materialized alternative versions so VSQL may be further confined to AV-SQL and RV-SQL.

Finally, these queries are executed and their results are merged by queuing them in temporary store. For optimization purposes *Inlining technique* [15] has been use with few modifications.

To compute final results of various mono-version queries over heterogeneous versions, it is required to merge the result set of RV-SQL, AV-SQL and V-SQL by using temporary store. The conception of temporary storage has already been used to merge results of various queries, and it has been found efficient and in good condition in [16]. The only inefficiency found for such purpose is excessive transfer to data, for which number of optimization techniques exists [15].

It is to be made clear that results of the real versions will be merged while the simulation result sets will be kept separate, which will help the user in identifying real world and simulated data. The manager have the ability to store metadata in catalog and keeping contact with smaller component called 'metadata manager' whose responsibility is to find and deliver appropriate set of metadata from MV catalog.

With these abilities of processing queries Version Manager will provide integrated and solo access along with flexible environment for simulating business scenarios for quality decision making.

VI. QUERY PROCESSING IN VM

Given the above architecture, in order to process the query transparent to the users, VM will retrieve data from various version/s and merge the related results. Fig.3 (b) shows query processing components. For this purpose following steps are performed:

a) Splitting multi-version query into mono-version queries, user will input query in TSQL form, independent of the existence of various versions. VM will divide this multi-version query on MVILS to smaller queries executable on versions. In case of VSQL, it is further mapped to real and materialized alternative versions.
b) Evaluation, of mono version queries is done in which each query is evaluated for its expected results i.e. the queries for which execution is either not possible or useless will be rejected [8].
c) Execution, after splitting multi-version query, mono-versions queries are send to respective version by using versioning metadata from MV-Catalog.
d) Monitoring, [8] has defined monitoring algorithm, whose responsibility is to monitor the execution of queries.
e) Store the result in temporary storage, the results of executions are stored and merged in the temporary storage, but it is not allowed to merge real world results, with SBS.

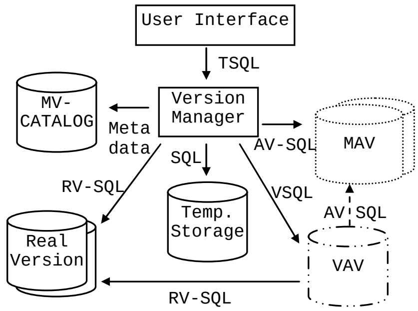


Fig.3. (a) Architecture of Version-Manager

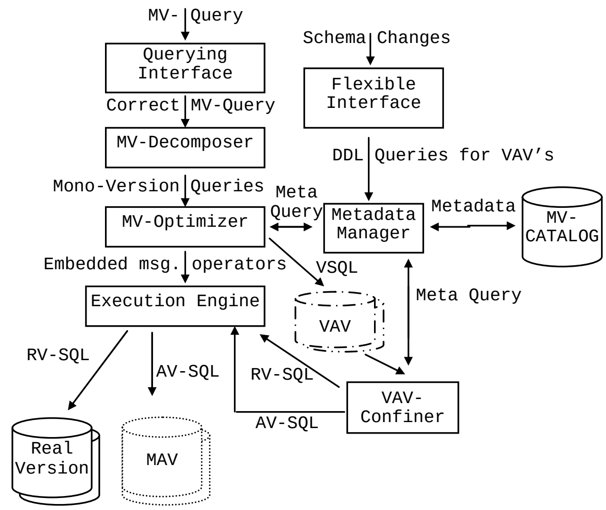


Fig. 3. (b) Query processing Components of Version-Manager

f) The results of various RV-SQL will be joined to form the reflection of the real world happenings.
g) Finally, if required additional grouping and/selection on the combined results is performed.

In the splitting step query may be divided into three parts:

a) RV-SQL executable on mono versions [8].
b) AV-SQL executable on materialized alternative versions.
c) VSQL which is executable on virtual versions for meeting the simulating requirements. As, it do not have any instance so it will be further divided into further queries executable on real and materialized alternative versions.

MV-Catalog is an integrated data dictionary maintained by making extensions to the conventional catalog; it has the abilities of storing versioning related information, especially virtual alternative version and their mapping to set of real and materialized alternative versions.

VII. DISCUSSION

To increase the precision of decision making, we have extended the concept of alternative business scenarios, by dividing it into two types called static and dynamic alternatives, so for retrieval and maintenance of such alternative versions we have presented flexible solution in the form of added component to Synthetic Warehouse Builder. Version Manager will provide flexible environment for achieving vital advantages of simulating business scenarios.

Future works will focus on identification and storing metadata items and architecture extensions to catalog for transparent querying. Also, maintaining consistency of results in the presence of concurrent execution of queries on multiple versions, and query maintenance algorithms for maintaining integrity algorithms are some more research issues.

NOTES

Extended version of the article is accepted for publication in Information Technology Journal.

This article is a direct contribution to the thesis '*achieving vital advantages of simulating business scenarios in multi-version data warehouse*' submitted to Royal Institute of Technology (KTH), Stockholm, Sweden.

The work was initiated at KTH and finalized at CIIT, Lahore.

REFERENCES

[1] Bebel, B. Eder, J. Koncilia, C. Morzy, T. Wrembel, R. Creation and Management of Versions in Multiversion Data Warehouse, Proceedings of *2004 ACM Symposium on Applied Computing*, 2004.

[2] Soberg, D. Quantifying Schema, *Evolution Information Software Technology* 35, 1, 35-54, 1993.

[3] Blaschka, M. Sapia, C. Hofling, G. On Schema Evolution in Multidimensional Databases. Proceedings of *DaWak99 Conference*, Italy, 1999.

[4] Hurtado, C.A. Mendelzon, A.O. Vaisman, A.A. Maintaining Data Cubes under Dimension Updates. Proceedings of *ICDE Conference*, Australia, 1999.

[5] Pasha, M.A. Nasir, J.A. Shahzad, M.K Semi-star schema for managing data warehouse consistency, In Proceedings of

This page intentionally blank

National Conference on Emerging Technologies, IEEE-ACM, NCET'04, Karachi, Pakistan, 2004.

[6] Wrembel R., Morzy T. Multiversion Data Warehouses: Challenges and Solutions. Proc. of the 3rd IEEE *Conference on Computational Cybernetics* (ICCC 2005), Mauritius, April 2005.

[7] Morzy T., Wrembel R.: On Querying Versions of Multi-version Data Warehouse. In Proceedings of *7th ACM DOLAP'04*, Washington, USA, 2004.

[8] Nasir, J.A. Shahzad, M.K. Pasha, M.A. Transparent querying multi-version data warehouse, *Information Technology Journal* 5 (2): 250-259, 2006.

[9] Shahzad,M.K. Recent issues identified and conceptualized for achieving vital advantages in multi-version data warehouse. *International Journal of Soft Computing* 1(2): 97-102, 2006.

[10] Eder, J. Koncilia, C. Changes of Dimension Data in Temporal Data Warehouses. Proceedings of *DaWak*, 2001.

[11] Eder, J. Koncilia, C. Morzy, T. The COMET Metamodel for temporal Data Warehouses, Proceedings of *CAiSE'02*, Canada, 2002.

[12] Chamoni, P. Stock, S. Temporal structures in Data Warehousing. Proceedings of the *DaWaK*, Italy, 1999.

[13] Mendelzon, A.O. Viasman, A.A. Temporal Queries in OLAP, Proceedings of *VLDB Conference*, Egypt, 2000.

[14] Bernadrino, J. Madera, H. Data Warehousing and OLAP: Improving Query Performance Using Distributed Computing, Proceedings of *CAiSE'00*, Sweden, 2000.

[15] Pedersen, D. Riis, K. Pedersen, T.B. Query optimization for OLAP XML federation, In Proceedings of *5th ACM Workshop on DOLAP*, USA, 2002.

[16] Pedersen, D. Pedersen, J. Pedersen, T.B. Integrating XML Data in Target OLAP system, In Proceedings of *9th ICDE*, 2004.

Intermingling evolutionary and versioning approach for data warehouse by Versioning-Algebra

M.Khurram.Shahzad, J.A.Nasir, M.A.Pasha
{mkshahzad, janasir,mapasha}@pucit.edu.pk
Intelligent Information Systems Research Group (RG-IIS),
Punjab University College of Information Technology,
University of the Punjab, Lahore.

ABSTRACT

Traditional databases are unable to support analytical business requirements. As a result, the conception of data warehouse was floated. So, data warehouse with its subjective schemas facilitates analytical business requirements. But, Conventional data warehouse cannot handle changes to its operational sources, for such purpose two approaches had been proposed: i) evolution ii) versioning.

In this study, we have presented a blend of evolution and versioning approaches with the help of schema-versioning-functions, named: i) versioning function ii) reviving function iii) qualifying function. The paper formalizes algebra for version evolution operations (VEO) by modifying existing calculi. This algebra will provide strong foundations for data warehousing tools which evolutes and maintains multiple versions.

KEYWORDS

Data Warehouse, Information system automation, Versioning Operation, Schema Versioning, Multi-dimensional Schema, Versioning-algebra.

1. INTRODUCTION

Traditional databases have normalized schemas, but analytical requirements have not found to be met by operational databases. So, the concept of data warehouse (DW) was floated, 'a data warehouse is a subject-oriented, time-variant, non-volatile, integrated collection of data in favor of decision-making.'[1, 6]. DW has been meeting analytical requirements for years, but depends on operational system/s, to meet subjective-data requirements.

DW reflects real world data, stored in subjective-form, optimized to support decision-making. Nevertheless, there are some changes in operational sources that result in derivation of inconsistent analytical results [2, 3]. Researchers of the domain are trying to devise reliable strategies, which support dynamic user's requirements, which can further be used for analytical purposes but no complete solution is available.

DW depends upon its data source/s, changes to these sources my result in unrealistic results, thus affecting decision-making to large extent. 'These changes can generally be categorized into two types]:

1. Content Changes: insertion /updation/ deletion of records occurred as a result of DML commands on database.
2. Schema Changes: addition/modification/ dropping of attribute occurred as a result of DDL commands on database.'[4]

Several schemas have been developed based on Star-oriented approach like: Star schema [5, p-210], Star-flake schema [6, p-144], Snowflake schema [5, p-238] and Constellation schema [6, p-218] has been forestalled for data warehouse but these changes cannot be handled. Alternative approach to the problem is evolution of schema [7], and versioning of DW [8].

Contributions -- In this paper we use calculi called versioning-algebra (V-algebra) to make a blend of evolutionary and versioning approaches, for maintaining DW under changes. The framework is based on three functions: i) versioning, to decide the evolution of schema or building new version ii) reviving function, decide the relationship between versions i.e. parent-child relationship iii) decides validity of versions. Formal operations, that result in either evolution or versioning of DW has already been defined by Blaschka [7] In our work, we have modified her schema paradigms and make the algebra support evolution and versioning in DW for operations.

The rest of paper is organizes as follows: Section 2 discusses example illustrating potential problem. Section

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 295–300.

3 overviews attempts made to solve the problem. Section 4 has multidimensional formalization paradigms, Section 5 gives schema-versioning function, and Section 6 presents versioning operations, while Section 7 gives A-algebra to cater changes. Paper will conclude by concluding the guideline for a tool.

2. ILLUSTRATIVE EXAMPLE

To better understand the problem, consider Eder's [8] example of a DW for company's sales, sale points in multiple cities. Sales are inspected in various locations at certain time. Cities are grouped into administrative regions and products are grouped into categories. Multi-dimensional schema (MD schema) of company's DW, defined by Eder. It is composed of three dimensions and a fact table, two of the three dimensions has two dimension tables. Amount is the only fact available in the fact table, sales_fact. Amount is calculated as product of Rate and Qty of the product. i.e.

Amt = (Sales_Fact . Qty) * (Category . Rate)

The tables store the following data. Here are the dummy values inserted to the tables along with the queries executed to retrieve the results.

Select * from **product**;
Prod_ID (1, 2, 3)
Prod_Name (P1,P 2,P 3)
Cat_ID (Category1, Category 2, Category 3)

Select * from **category**;
Cat_ID (1, 2, 3)
Cat_Name (Category1, Category2, Category 3)
Rate (Category1, Category 2, Category 3)

Select * from **region**;
Region_ID (1, 2)
Region_Name (Region A, Regsion B)

Select * from **city**;
City_ID (1, 2, 3)
City_Name (BWP, ISL, KHI)
Region_ID (1, 2, 1)

Select * from **Sales_Fact**;

Time_ID	Prod_ID	City_ID	Qty	Amt
1	1	1	65	650
1	2	1	90	900
1	3	2	32	3200
1	1	3	20	2000

Eder illustrated problems related to the retrieval of inconsistent analytical results due to content changes i.e. changing the borders of regions may results in moving cities from one region to another. Such a change may have an impact on the analytical results received from DW. Assume that boundaries of region are changed in such a way that city "KHI" is moved from "Region A" to "Region B".

Let us assume a query-computing amount earned in every region before changing the boundaries of region is:
Region_Name (Region A, Region B)
Sum (Amount) (3550, 3200)

And the total amount earned in every region after changing the boundary of region is:
Region_Name (Region A, Region B)
Sum (Amount) (1550, 5200)

Firstly, as proved by example inconsistent results are produced due to content changes. Similarly, inconsistent results are also generated if region name is changed, city name is changed, product category is changed, product name is changed, and name of product category is changed. Also, when the schema is upgraded this may not give desired results.

Secondly, adding new attribute/s in one of the sources may require adding this attribute to the MD-schema if one would like to analyze values of this attribute in the DW. Also it is possible that this contributes to the calculation of facts, convectional MD-schema cannot handle these changes.

3. MD-SCHEMA FORMALIZATION PARADIGM

Two major attempts have been made, to handle dynamics to operational source [8]. These are i) Schema evolution ii) Schema versioning. First approach maintains only one schema; changes are applied to that schema, Blaschka [7] has presented evolution method with the help of algebra. In second approach changes are applied to new schema and both versions are maintained.

For our research work we need formalism that can serve as a basis for defining the schema versioning operations and their effects. Therefore, this section contains formal-mathematical notations for MD schema definition. Paradigm defined for MD-schema and model of Blaschka *et al.* [7] has been modified, by different set of attributes, to be used for versioning.

Here we assume a universal set U of alphabets to represent versions, its validity time, set of facts, dimension levels, attribute names with granularity level. So paradigm for it can be defined as:

Definition 1 – Versioned MD-Schema
Each version of MD-schema (ℵ) has seven attributes, defined as:

$\aleph < V_{x,y}, VT_m, F_k, L_j, A_t, \textit{gran}, \textit{attr} >$

Where

- $V_{x,y} \subset U$ is finite set of versions of DW schema $\{V_{0,1}, V_{1,1} V_{2,1} \ldots V_{m,n}\}$ where, $V_{x,y} \in U$ for $0 \le x \ge m$ and $0 \le y \ge n$.
- VT_m is valid time of real and alternative version calculated by function $\wp$ i.e.
  $f(VT_m) \rightarrow VT_p \{VT_r, VT_a\}$
  $g(VT_p) \rightarrow S \{0, 1\}$
  $\wp(VT_m) : g \circ f(VT_m) \rightarrow X \{0,1\}$
- $F_k \subset U$ is finite set of fact names $\{F1, F_2, \ldots F_m\}$ where, $F_k \in U$ for $0 \le k \ge m$.*
- $L_j \subset U$ is finite set of dimension level names $\{L_1, L_2, \ldots L_n\}$ where, $L_j \in U$ for $0 \le j \ge n$.*
- $A_t \subset U$ is finite set of attribute names $\{A_1, A_2, \ldots A_n\}$ where, $A_t \in U$ for $0 \le t \ge n$.*
- *gran:* $F \rightarrow 2^L$ is a function that associates a fact with a set of dimensional level names. These dimension levels gran (f) are called the base level of fact f.*
- *attr:* $A \rightarrow F \cup L \cup \{\perp\}$ is a function mapping an attribute either to a fact, to a dimension level or a special $\perp$ symbol which means that this attribute is not connected alt all.*

It is to be noted that attributes with (*) at the end are same as produced by Blaschka [7].

4. SCHEMA VERSIONING FUNCTION

Our approach to the problem is based on V-algebra, so in order to objectively handle problem-generating changes mentioned earlier in this paper, we have defined a versioning-function (ϑ) with the ability of versioning or evolution of MD-schema. This function executes as a result of operations on MD-schema.

Definition 2 – Versioning Function

A function which take MD-schema as input argument and produces new MD-schema due to operation $Oper_a$ on $\aleph$. Mathematically, this function is defined as:

$$\aleph_n : Oper_a(\aleph_o)$$

this operation results making changes to version also i.e.

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$\aleph_{evol}$ means schema is evaluated, i.e. changes are made to existing schema without maintaining new version.

$\aleph_{vers}$ means new version of schema is created and maintained.

Versioning-function is executed as a result of set of formal versioning-operations defined in next section. As versioning function is applied to MD-schema, which results in either evolution of schema or creation of new version. If schema evolutes then $V_{x,y}$ i.e. version identifier is not changed, but other values may change depending upon change made. Else if new version is created, $V_{x,y}$ along with other attributes.

Definition 2 – Reviving Function

A function which takes MD-schema version identifier as input argument and concludes it depending reviving function i.e. changing the values of x and y defined as:

$$\Psi(V_{x,y}) : \begin{cases} \text{if } \vartheta(V_{x,y}) = V_{vers} \\ \quad \begin{cases} x = x \\ \& \\ y = y \end{cases} \\ \text{Else} \\ \quad \begin{cases} x = x.i; & \text{for AV.} \\ \text{i is AV identifier} & \\ x = x + 1 & \text{else} \end{cases} \\ y = \text{parent version identifier.} \end{cases}$$

Definition 3 – Qualifying Function

After taking MD-schema version identifier qualifies it for further use, by changing value of VT_m attribute of identifier is defined as:

$\wp(VTm) : g \circ f(VTm) \rightarrow X \{0,1\}$
$f(VTm) \rightarrow VTp \{VTr, VTa\}$
$g(VTp) \rightarrow S \{0, 1\}$

5. VERSIONING OPERATIONS (VO's)

After having formally defined: a) mathematical model for MD-schema and b) versioning-function, we are going to present set of operations which results in either evolution or versioning of schema but with different attributes of $\aleph$ i.e. values of $V_{x,y}$, VT_m, F_k, L_j, A_t, *gran*, *attr* can change.

The operations are [7], inserting or deleting fact table, inserting or deleting dimension level, connecting or disconnecting attribute with dimension level, insertion or deletion of fact, new attribute insertion or deletion, connecting or disconnecting attributes to fact, inserting or deleting dimension.

6. CATTERING VO's BY V-ALGEBRA

In order to be able to manage problem-creating changes a model of multi-versioned DW has been developed. In this approach, changes to a schema may be applied to a new version called 'Child Version' which is explicitly derived from pervious version, called 'Parent Version'.

1. ***Level insertion:*** it extends schema by inserting new dimension level. All this, results in changed parameters of MD-Schema.

Set before change

$\aleph_o < V_{x,y}, VT_m, F_k, L_j, A_t, gran, attr >$

Set is change due to operation $Oper_{li}$

$\aleph_n : Oper_{li}(\aleph_o)$

i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j', A_t, gran', attr' >$

where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$L_j' : L \cup \{l_{new}\}$

$gran' : gran(f) ; gran(f) : F \rightarrow 2^{L'}$

$VT_m' : \wp(VT_m) ;$

$\wp(VTm) : g \circ f(VTm) \rightarrow X \{0,1\}$

$f(VTm) \rightarrow VTp \{ VTr, VTa \}$

$g(VTp) \rightarrow S \{0, 1\}$

$attr' : attr(a) ; attr(a) : A \rightarrow F \cup L' \cup \{\perp\}$

In all coming points set before change, changes to version i.e. $V_{x,y}$ and its validity time VT_m will change in the same way as this one.

In all coming points set before change, changes to version i.e. $V_{x,y}$ and its validity time VT_m will change in the same way as this one. It is important to note that, the attributes whose affect is same as Blaschka's work are taken from [7].

2. ***Level Deletion:*** it extends schema by deleting dimension level. Mathematically, operation removes an element from set of levels. All this results in changes in parameters of MD-Schema.

Set is change due to operation $Oper_{ld}$

$\aleph_n : Oper_{ld}(\aleph_o)$

i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j', A_t, gran', attr' >$

where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m) ;$

$\wp(VTm) : g \circ f(VTm) \rightarrow X \{0,1\}$

$f(VTm) \rightarrow VTp \{ VTr, VTa \}$

$g(VTp) \rightarrow S \{0, 1\}$

$L_j' : L \cup \{l_{del}\}$

$gran' : gran(f) ; gran(f) : F \rightarrow 2^{L'}$

$attr' : attr(a) ; attr(a) : A \rightarrow F \cup L' \cup \{\perp\}$

3. ***Insert attribute:*** creates a new attribute without attaching it to a dimension level or fact, as assigning an existing attribute to dimension level or fact is a separate operation i.e. attributed connection.

Set is change due to operation $Oper_{ia}$

$\aleph_n : Oper_{ia}(\aleph_o)$

i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t', gran, attr' >$

where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m) ;$

$\wp(VTm) : g \circ f(VTm) \rightarrow X \{0,1\}$

$f(VTm) \rightarrow VTp \{ VTr, VTa \}$

$g(VTp) \rightarrow S \{0, 1\}$

$A_t' : A \cup \{a_{new}\}$

$attr' : attr(a) ; attr(a) : A \rightarrow F \cup L' \cup \{\perp\}$

4. ***Delete attribute:*** creates a new attribute without attaching it to a dimension level or fact, as assigning an existing attribute to dimension level or fact is a separate operation i.e. attributed connection.

Set before change is $\aleph_o$ as shown above.

Set is change due to operation $Oper_{da}$

$\aleph_n : Oper_{da}(\aleph_o)$

i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t', gran, attr' >$

where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m) ;$

$\wp(VTm) : g \circ f(VTm) \rightarrow X \{0,1\}$

$f(VTm) \rightarrow VTp \{ VTr, VTa \}$

$g(VTp) \rightarrow S \{0, 1\}$

$A_t' : A \cup \{a_{new}\}$

$attr' : attr(a) ; attr(a) : A \rightarrow F \cup L' \cup \{\perp\}$

5. ***Connect attribute to dimension level:*** Connects an existing attribute a_{new} to a dimension level $l \in L$ and $a_{new} \in A$.

Set is change due to operation $Oper_{cdl}$

$\aleph_n : Oper_{cdl}(\aleph_o)$

i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran, attr' >$

$V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$
$f(VTm) \to VTp\ \{VTr, VTa\}$
$g(VTp) \to S\ \{0, 1\}$

$$attr' : \begin{cases} l & \text{if } a = a_{new} \\ attr(a) & \text{else} \end{cases}$$

$attr(a) : A \to F \cup L \cup \{\perp\}$

6. ***Disconnect attribute to dimension level:*** Disconnects an existing attribute a_{new} from dimension level $l \in L$.
Set is change due to operation $Oper_{dcdl}$
$\aleph_n : Oper_{dcdl}(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran, attr' >$
$V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \to \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$
$f(VTm) \to VTp\ \{VTr, VTa\}$
$g(VTp) \to S\ \{0, 1\}$

$$attr' : \begin{cases} \perp & \text{if } a = a_{new} \\ attr(a) & \text{else} \end{cases}$$

$attr(a) : A \to F \cup L \cup \{\perp\}$

7. ***Connect attribute to fact:*** connects an existing attribute a_{new} to fact $f \in F$.
Set is change due to operation $Oper_{caf}$
$\aleph_n : Oper_{caf}(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran, attr' >$
where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \to \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$
$f(VTm) \to VTp\ \{VTr, VTa\}$
$g(VTp) \to S\ \{0, 1\}$

$$attr' : \begin{cases} f & \text{if } a = a_{new} \\ attr(a) & \text{else} \end{cases}$$

$attr(a) : A \to F \cup L \cup \{\perp\}$

8. ***Disconnect attribute from fact:*** disconnects an existing attribute a_{new} to fact $f \in F$.
Set is change due to operation $Oper_{dcaf}$
$\aleph_n : Oper_{dcaf}(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran, attr' >$
Where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \to \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$
$f(VTm) \to VTp\ \{VTr, VTa\}$
$g(VTp) \to S\ \{0, 1\}$

$$attr' : \begin{cases} \perp & \text{if } a = a_{new} \\ attr(a) & \text{else} \end{cases}$$

$attr(a) : A \to F \cup L \cup \{\perp\}$

9. ***Insert fact:*** this operation extends the MD-model by a new fact. The operation extends the set of facts without attaching dimension levels to this fact Dimensions for this fact have to defined separately.
Set is change due to operation $Oper_{if}$
$\aleph_n : Oper_{if}(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k', L_j, A_t, gran', attr' >$
where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \to \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$
$f(VTm) \to VTp\ \{VTr, VTa\}$
$g(VTp) \to S\ \{0, 1\}$
$F_k' : F_k \cup \{f_{new}\}$
$attr' : attr(a)$; $attr(a) : A \to F \cup L' \cup \{\perp\}$

$$gran' : \begin{cases} \emptyset & \text{if } f = f_{new} \\ gran(f) & \text{else} \end{cases}$$

$gran(f) : F' \to 2^{L'}$

10. ***Delete fact:*** this operation reduces MD-model by one fact. The fact must not be connected to a dimension ($gran(f_{del}) = \emptyset$) and must not contain any attributes ($attr(a) \neq f_{del}$ for all $a \in A$).
Set is change due to operation $Oper_{df}$
$\aleph_n : Oper_{df}(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k', L_j, A_t, gran', attr' >$
where $V_{x,y}' : \vartheta(V_{x,y})$

$$\vartheta(V_{x,y}) \to \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)$;
$\wp(VTm) : g \circ f(VTm) \to X\ \{0,1\}$

$f\,(VTm) \rightarrow VTp\ \{ VTr\ , VTa \}$
$g\,(VTp) \rightarrow S\ \{0, 1\}$
$F_k' : F_k \setminus \{f_{del}\}$
$attr' : attr\,(a)\ ;\ attr\,(a) : A \rightarrow F \cup L' \cup \{\perp\}$

$$gran' \rightarrow \begin{cases} \varnothing & \text{if } f = f_{new} \\ \text{gran}\,(f) & \text{else} \end{cases}$$

11. ***Insert Dimension:*** inserts a dimension from a fact.

Set is change due to operation $Oper_{id}$
$\aleph_n : Oper_{id}\,(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran', attr >$
where $V_{x,y}' : \vartheta\,(V_{x,y})$

$$\vartheta\,(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)\,;$
$\wp(VTm) : g \text{ o } f\,(VTm) \rightarrow X\ \{0,1\}$
$f\,(VTm) \rightarrow VTp\ \{ VTr\ , VTa \}$
$g\,(VTp) \rightarrow S\ \{0, 1\}$

$$gran' : \begin{cases} \text{gran (f)} & \text{if } f \neq f_{ins} \\ \text{gran}\,(f) \cup \{l\} & \text{if } f = f_{ins} \end{cases}$$

$gran\,(f) : F \rightarrow 2^L$

12. ***Delete Dimension:*** deletes a dimension from a fact.

Set is change due to operation $Oper_{dd}$
$\aleph_n : Oper_{dd}\,(\aleph_o)$
i.e. $\aleph_n < V_{x,y}', VT_m', F_k, L_j, A_t, gran', attr' >$
where $V_{x,y}' : \vartheta\,(V_{x,y})$

$$\vartheta\,(V_{x,y}) \rightarrow \begin{cases} V_{evol} & \text{if MD-schema evolutes} \\ V_{vers} & \text{else schema is versioned} \end{cases}$$

$VT_m' : \wp(VT_m)\,;$
$\wp(VTm) : g \text{ o } f\,(VTm) \rightarrow X\ \{0,1\}$
$f\,(VTm) \rightarrow VTp\ \{ VTr\ , VTa \}$
$g\,(VTp) \rightarrow S\ \{0, 1\}$

$$gran' : \begin{cases} \text{gran (f)} & \text{if } f \neq f_{del} \\ \text{gran}\,(f) - \{l\} & \text{if } f = f_{del} \end{cases}$$

$gran\,(f) : F \rightarrow 2^L$

7. CONCLUSION

Multiple versions of DW are required to be maintained, to meet business dynamics. In this study we have modified Balaschka's work to make it applicable for evolution or versioning. For this work we have: a) firstly, defined multi-dimensional schema b) secondly, schema versioning functions c) Finally, based on formal versioning-operations, we have defined the affects the affects of these operations on multi-dimensional schema. Similarly, retrieval of analytical results from multiple versions, documenting multiple versions, data management techniques and efficiency of processing are some future research directions.

REFERENCES

[1] Jarke,M. Lenzerini,M. Vallilious,Y. *Fundamentals of data warehouses*. Springer Verlag. 2000.

[2] Wrembel R., Morzy T. *Multiversion Data Warehouses: Challenges and Solutions*. Proc. of the 3rd IEEE Conference on Computational Cybernetics (ICCC 2005), Mauritius, April 2005.

[3] Rundensteiner, E. Koeller, A. Xhang, X. *Maintaining Data Warehouse over changing information sources*, Communications of the ACM, Vol. 43, No.6, 2000.

[4] Pasha, M.A. Nasir, J.A. Shahzad, M.K *Semi-Star Schema for Managing Data Warehouse Consistency,* Proceedings of IEEE-ACM-NCET, 2004.

[5] Paulraj,P. *Data Warehousing fundamentals: A comprehensive guide for I.T Professionals*, John Wiley & Sons, 2001.

[6] Inmon,W.H. *Building the Data Warehouse*, 2nd Edition, John Wiley & Sons, 2001.

[7] Blaschka, Carsten Sapia, Hofling, *On Schema Evolution in Multi-dimensional Databases*, Proceedings of DaWak, 1999

[8] Bebel, B. Eder, J. Koncilia, C. Morzy, T. Wrembel, R. *Creation and Management of Versions in Multiversion Data Warehouse*, Proceedings of 2004 ACM Symposium on Applied Computing, 2004.

An efficient fault-tolerant scheduling algorithm for precedence constrained tasks in heterogeneous distributed systems

M. Nakechbandi[*], J.-Y. Colin[*], J.B. Gashumba[**]

LITIS Université du Havre, 25 rue Philippe Lebon, BP 540, 76058, Le Havre cedex, France
[(*)]{moustafa.nakechbandi, jean-yves.colin}@univ-lehavre.fr
[(**)] jb.gashumba@free.fr

Abstract
In this paper, we propose an efficient scheduling algorithm for problems in which tasks with precedence constraints and communication delays have to be scheduled on an heterogeneous distributed system with one fault hypothesis. Our algorithm combines the DSS_OPT algorithm and the eFRCD algorithm. To provide a fault-tolerant capability, we employ primary and backup copies. In this scheme, backup copies can overlap other backup copies, and backup copies are not established for tasks that have more than one primary copy. The result is a schedule in polynomial time that is optimal when there is no failure, and is a good resilient schedule in the case of one failure.

Keywords: Scheduling, Heterogeneous Distributed System, Fault-tolerance, Precedence Constraints, Communication Delays, Critical-Path Method.

I. INTRODUCTION

Heterogeneous distributed systems have been increasingly used for scientific and commercial applications, some of the current ones include Automated Document Factories (ADF) in banking environments where several hundred thousands documents are produced each day on networks of several multiprocessors servers, or high performance Data Mining (DM) systems [10] or Grid Computing [9, 11]. However, using efficiently these heterogeneous systems is a hard problem, because the general problem of optimal scheduling of tasks is NP-complete [13].

When the application tasks can be represented by Directed Acyclic Graphs (DAGs), many dynamic scheduling algorithms have been devised. For some examples, see [2, 3, 7]. Also, several static algorithms for scheduling DAGs in meta-computing systems are described in [1, 4, 6, 13]. Most of them suppose that tasks compete for limited processor resources, and thus these algorithms are mostly heuristics. In [5] is presented an optimal polynomial algorithm that schedules the tasks and communications of an application on a Virtual Distributed System with several clusters' levels, although, in [8] is studied a static scheduling problem where the tasks execution times are positive independent random variables, and the communication delays between the tasks are perfectly known.

This review shows that a lot of works has been done concerning heterogeneous distributed scheduling.

However, the problems with fault tolerant aspect are less studied. Reliable execution of a set of tasks is usually achieved by task duplication and backup copy [16, 17, 18]. Inspired with the works of Xiao Qin, Hong Jiang on Real-Time Heterogenous Systems [16], we propose in this article a good algorithm to solve the problem of one fault tolerant system using tasks duplication and backup copies technics.

This paper has three main parts: In the first one, we present the problem, in the second part, we present a solution to the problem, and in the third part, we discuss the advantages and disadvantages of the proposed solution.

II. THE CENTRAL PROBLEM

In this part, we present the following problem : Given an application T, we want to build an off-line scheduling of the tasks of this application in an heterogeneous distributed System with some possibilities of limited failures. We suppose that only one permanent failure can occur without any possibility of a temporary failure. This constitute one hypothesis which we call "1-failure". Note that the processing time of the application has to be minimized.

2.1 The Distributed Servers System

We call Distributed Servers System (DSS) a virtual set of geographically distributed, multi-users, heterogeneous or not, servers. Therefore, a DSS has the following properties:
First, the processing time of a task on a DSS may vary from a server to another. This may be due to the processing power available on each server of the DSS for example. The processing time of each task on each server is supposed known. Second, although it may be possible that some servers of a DSS are potentially able to execute all the tasks of an application, it may also be possible in some applications that some tasks may not be executed by all servers. This could be due to the fact that specific hardware is needed to process these tasks and that this hardware is not available on some servers. Or it could be that some specific data needed to compute these tasks are not available on these servers for some reason. Or it could be that some user input is needed and the user is only located in a geographically specific place. Obviously, in our problem we suppose that the needs of each task of an application are known, and that at least one server of the DSS may process it, else there is no possible solution to the scheduling problem.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 301–307.

Furthermore, an important hypothesis is that the concurrent executions of some tasks of the application on a server have a negligible effect on the processing time of any other task of the application on the same server. Although apparently far-fetched, this hypothesis may hold if the server is a multiprocessors architecture with enough processors to simultaneously execute all the tasks of the application that are to be processed concurrently. Or it may be that the server is a time-shared, multi-user system with a permanent heavy load coming from other applications, and the tasks of an application on this server represent a negligible additional load compared to the rest.

In addition, in the network interconnecting the servers of a DSS, the transmission delay of a result between two tasks varies depending on the tasks and on their respective sites.

Again, we suppose that concurrent communications between tasks of the same application on two servers have a negligible effect on the communication delays between two others tasks located on the same two servers. This hypothesis may hold if the network already has a permanent heavy load due to other applications, and the communications of the application represent a negligible additional load compared to the one already present.

2.2 Directed Acyclic Graph

We now describe the application itself in our problem. An application is decomposed into a set of indivisible tasks that have to be processed. A task may need data or results from other tasks to fulfil its function and then send its results to other tasks. The transfers of data between the tasks introduce dependencies between them. The resulting dependencies form a Directed Acyclic Graph. Because the servers are not necessarily identical, the processing time of a given task can vary from one server to the next. Furthermore, the duration of the transfer of a result on the network cannot be ignored. This communication delay is function of the size of the data to be transferred and of the transmission speed that the network can provide between the involved servers. Note that if two dependent tasks are processed themselves on the same server, this communication delay is considered to be 0.

The central scheduling problem *P* on a Distributed Server System, is represented therefore by the following parameters:

- a set of servers, noted $\Sigma = \{\sigma_1, ..., \sigma_s\}$, interconnected by a network,
- a set of the tasks of the application, noted $I = \{1,..., n\}$, to be executed on Σ. The execution of task i, $i \in I$, on server σ_r, $\sigma_r \in \Sigma$, is noted i/σ_r. The subset of the servers able to process task i is noted Σ_i, and may be different from Σ,
- the processing times of each task *i* on a server σ_r is a positive value noted π_{i/σ_r}. The set of processing times of a given task *i* on all servers of Σ is noted $\Pi_i(\Sigma)$. $\pi_{i/\sigma_r} = \infty$ means that the task *i* cannot be executed by the server σ_r.
- a set of the transmissions between the tasks of the application, noted U. The transmission of a result of an task *i*, $i \in I$, toward a task *j*, $j \in I$, is noted (i, j). It is supposed in the following that the tasks are numbered so that if $(i, j) \in U$, then $i < j$,
- the communication delays of the transmission of the result (i, j) for a task i processed by server σ_r toward a task j processed by server σ_p is a positive value noted $c_{i/\sigma_r, j/\sigma_p}$. The set of all possible communication delays of the transmission of the result of task i, toward task j is noted $\Delta_{i,j}(\Sigma)$. Note that a zero in $\Delta_{i,j}(\Sigma)$ mean that i and j are on the same server, i.e. $c_{i/\sigma_r, j/\sigma_p} = 0 \Rightarrow \sigma_r = \sigma_p$. And $c_{i/\sigma_r, j/\sigma_p} = \infty$ means that either task *i* cannot be executed by server σ_r, or task *j* cannot be executed by server σ_p, or both.

Let $\Pi(\Sigma) = \bigcup_{i \in I} \Pi_i(\Sigma)$ be the set of all processing times of the tasks of *P* on Σ.

Let $\Delta(\Sigma) = \bigcup_{(i,j) \in U} \Delta_{i,j}(\Sigma)$ be the set of all communication delays of transmissions (i, j) on Σ.

The central scheduling problem *P* on a distributed servers system DSS can be modelled by a multi-valued DAG $G = \{I, U, \Pi(\Sigma), \Delta(\Sigma)\}$. In this case we note $P=\{G, \Sigma\}$.

Example: In the following example we consider a problem *P* of an application with nine tasks that has to be processed by a set of 3 servers on an heterogeneous distributed system. The architecture of distributed system and the task graph of this application are represented by the figure 1 and figure 2. It is supposed that the 3 servers are distributed on two different networks.

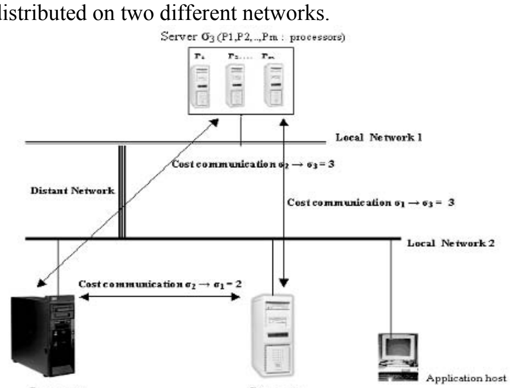


***Figure 1**: Distributed system architecture for the application example.*

The Matrix cost communication of our example is presented in table 1.

Network delay between $\sigma_i \to \sigma_j$	Server σ_1	Server σ_2	Server σ_3
Server σ_1	0	2	3
Server σ_2	2	0	3
Server σ_3	3	3	0

***Table 1**:* Cost communication *between servers* (d*istance* $\sigma_r \to \sigma_p$ *)*

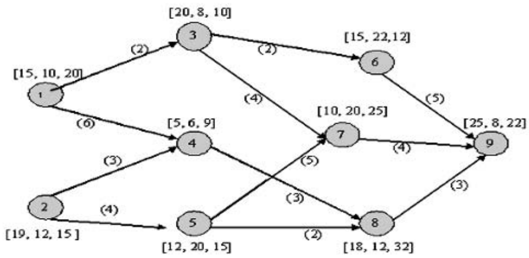


Figure 2: *Example of a multi-valued DAG*

In this example there are 9 tasks, the label [x, y, z] on task is the processing cost on the 3 servers'. For example on the task 1 we have the label [15, 10, 20], that is mean the processing time of task 1 on server σ_1 is 15, on server 2 is 10 on server 3 is 20. The table 2 indicate the complete communication delays for the problem *P*, this effective communication delay between two tasks is due to the multiplication of cost communication (data in tables 1) and volume communication data between tow tasks. For example, if task 1 is executed on server σ_1 , task 3 is executed on server σ_2 the communication between tasks 1 and 3 noted $c_{1/\sigma_1,3/\sigma_2}$ = 2*2 = 4, because the cost communication ($\sigma_1 \rightarrow \sigma_2$) is 2, the volume of data between task1 and task 3 is 2 also.

(1, 3)	σ_1	σ_2	σ_3
σ_1	0	4	6
σ_2	4	0	6
σ_3	6	6	0

(3, 6)	σ_1	σ_2	σ_3
σ_1	0	4	6
σ_2	4	0	6
σ_3	6	6	0

(5, 8)	σ_1	σ_2	σ_3
σ_1	0	4	6
σ_2	4	0	6
σ_3	6	6	0

(1, 4)	σ_1	σ_2	σ_3
σ_1	0	12	18
σ_2	12	0	18
σ_3	18	18	0

(3, 7)	σ_1	σ_2	σ_3
σ_1	0	8	12
σ_2	8	0	12
σ_3	12	12	0

(6, 9)	σ_1	σ_2	σ_3
σ_1	0	10	15
σ_2	10	0	15
σ_3	15	15	0

(2, 4)	σ_1	σ_2	σ_3
σ_1	0	6	9
σ_2	6	0	9
σ_3	9	9	0

(4, 8)	σ_1	σ_2	σ_3
σ_1	0	6	9
σ_2	6	0	9
σ_3	9	9	0

(7, 9)	σ_1	σ_2	σ_3
σ_1	0	8	12
σ_2	8	0	12
σ_3	12	12	0

(2, 5)	σ_1	σ_2	σ_3
σ_1	0	8	12
σ_2	8	0	12
σ_3	12	12	0

(5, 7)	σ_1	σ_2	σ_3
σ_1	0	10	15
σ_2	10	0	15
σ_3	15	15	0

(8, 9)	σ_1	σ_2	σ_3
σ_1	0	6	9
σ_2	6	0	9
σ_3	9	9	0

Table 2: *Complete communication times for the problem P*

2.3 Definition of a feasible solution

We note PRED(*i*), the set of the predecessors of task *i* in *G*:

$$\text{PRED}(i) = \{\ k / k \in I \text{ et } (k,i) \in U\ \}$$

And we note SUCC(*i*), the set of the successors of task *i* in *G*: $\text{SUCC}(i) = \{\ j / j \in I \text{ et } (i,j) \in U\ \}$

A feasible solution S for the problem P is a subset of executions { i/σ_r , $i \in I$ } with the following properties:

- each task *i* of the application is executed at least once on at least one server σ_r of Σ_i,
- to each task *i* of the application executed by a server σ_r of Σ_i, is associated one positive execution date t_{i/σ_r} ,
- for each execution of a task *i* on a server σ_r, such that PRED(i) $\neq \varnothing$, there is at least an execution of a task *k*, $k \in$PRED(*i*), on a server σ_p, $\sigma_p \in \Sigma_\kappa$, that can transmit its result to server σ_r before the execution date t_{i/σ_r} .

The last condition, also known as the Generalized Precedence Constraint (GPC) [5], can be expressed more formally as:

$$\forall i/\sigma_r \in S \begin{cases} t_{i/\sigma_r} \geq 0 & \text{if } \text{PRED}(i) = \varnothing \\ \forall k \in \text{PRED}(i), \exists \sigma_p \in \Sigma_k / t_{i/\sigma_r} \geq t_{k/\sigma_p} + \pi_{k/\sigma_p} + c_{k/\sigma_p, i/\sigma_r} & \text{else} \end{cases}$$

It means that if a communication must be done between two scheduled tasks, there is at least one execution of the first task on a server with enough delay between the end of this task and the beginning of the second one for the communication to take place. A feasible solution *S* for the problem *P* is therefore a set of executions i/σ_r of all i tasks, $i \in I$, scheduled at their dates t_{i/σ_r} , and verifying the Generalised Precedence Constraints GPC. Note that, in a feasible solution, several servers may simultaneously or not execute the same task. This may be useful to generate less communications. All the executed tasks in this feasible solution, however, must respect the Generalized Dependence Constraints.

2.4 Optimality Condition

Let *T* be the total processing time of an application (also known as the makespan of the application) in a feasible solution *S*, with *T* defined as:

$$T = \max_{i/\sigma_r \in S} (t_{i/\sigma_r} + \pi_{i/\sigma_r})$$

A feasible solution *S** of the problem *P* modelled by a DAG *G* = {*I*, *U*, *Π*(Σ), *Δ*(Σ)} is optimal if its total processing time *T** is minimal. That is, it does not exist any feasible solution *S* with a total processing time *T* such that *T* < *T**.

III. PROPOSAL SOLUTION

Our algorithm has two phases: the first one is for the scheduling of primary copies where we use the DSS-OPT algorithm [15] and the second one is for the scheduling of the backup copies in which a variant of the eFRCD algorithm [16] is used.

3.1. Primary Copies Scheduling (The DSS_OPT Algorithm):

We schedule primary copies of tasks in our algorithm with DSS-OPT algorithm [15]. The DSS-OPT algorithm is an extension of PERT algorithms type. In its first phase, it computes the earliest feasible execution date of each task on every server, and in its second phase it builds a feasible solution (without fault) starting from the end of the graph with the help of the earliest dates computed in the first phase.

Let *P* be a DSS scheduling problem, and let *G* = {*I*, *U*, *Π*(Σ), *Δ*(Σ)} be its DAG.

One can first note that there is an optimal trivial solution to this DSS scheduling problem. In this trivial solution, all possible tasks are executed on all possible servers, and their results are then broadcasted to all other tasks that may need them on all others servers. This is an obvious waste of processing power and communication resources, however, and something better and more efficient is usually needed.

So, we now present DSS_OPT(P) algorithm's that builds an optimal solution for problem P.

DSS_OPT has two phases:
The first phase, DSS_LWB(P), computes the earliest feasible execution dates b_{i/σ_r} for all possible executions i/σ_r of each task i of problem P.

The second phase determines, for every task i that does not have any successor in P, the execution i/σ_r ending at the earliest possible date b_{i/σ_r}. If several executions of task i end at the same smallest date b_{i/σ_r}, one is chosen, arbitrarily or using other criteria of convenience, and kept in the solution. Then, for each kept execution i/σ_r that has at least one predecessor in the application, the subset L_i of the executions of its predecessors that satisfy GPC(i/σ_r) is established. This subset of executions of predecessors of i contains at least an execution of each of its predecessors in G. One execution k/σ_p of every predecessor task k of task i is chosen in the subset, arbitrarily or using other criteria of convenience, and kept in the solution. It is executed at date b_{k/σ_p}. The examination of the predecessors is pursued in a recursive manner until the studied tasks do not present any predecessors in G. The complete algorithm is the following

Input: $G = \{I, U, \Pi(\Sigma), \Delta(\Sigma)\}$
Output: A feasible solution

DSS_OPT (*P*)
DSS_LWB (P) // first phase
$T = \max_{\forall i/\mathrm{SUCC}(i)=\varnothing} \min_{\forall \sigma_r \in \Sigma_i} (r_{i/\sigma_r})$
for all tasks i such that SUCC(i) = ∅ // second phase
$L_i \leftarrow \{ i/\sigma_r / \sigma_r \in \Sigma_t$ and $r_{i/\sigma_r} \le T \}$
$i/\sigma_r \leftarrow$ keepOnefrom(L_i)
schedule (i/σ_r)
end DSS_OPT

DSS_LWB(*P*)
For each task i where PRED(i) = ∅ **do**
for each server σ_r such that $\sigma_r \in \Sigma_i$ **do**
$b_{i/\sigma_r} \leftarrow 0$
$r_{i/\sigma_r} \leftarrow \pi_{i/\sigma_r}$
end for
mark (i)
end for

while there is a non marked task i such that all its predecessors k in G are marked **do**

for each server σ_r such that $\sigma_r \in \Sigma_i$ **do**
$b_{i/\sigma_r} \leftarrow \max_{\forall k \in \mathrm{PRED}(i)} \min_{\forall \sigma_p \in \Sigma_k} (b_{k/\sigma_p} + \pi_{k/\sigma_p} + c_{k/\sigma_p, i/\sigma_r})$
$r_{i/\sigma_r} \leftarrow b_{i/\sigma_r} + \pi_{i/\sigma_r}$
end for
mark (i)
end while

end DSS_LWB(*P*)

schedule(*i*/σ_r)
execute the task i at the date b_{i/σ_r} on the server σ_r
if PRED(i) ≠ ∅ then
for each task k such that $k \in$ PRED(i) **do**
$L_k^{i/\sigma_r} \leftarrow \{ k/\sigma_q \ / \ \sigma_p \in \Sigma_K$ and $b_{k/\sigma_p} + \pi_{k/\sigma_p} + c_{k/\sigma_p, i/\sigma_r} \le b_{i/\sigma_r} \}$
$k/\sigma_q \leftarrow$ keepOneFrom(L_k^{i/σ_r})
schedule (k/σ_q)
end for
end if
end schedule

keepOneFrom(L_i)
return an execution i/σ_r of task i in the list of the executions L_i.
end keepOneFrom.

Because the computed execution time of each task on each server is its earliest execution time on this server, and because only the copy with the ealiest ending time, of each task without any successor, is used in the solution calculated by DSS_OPT, and finally because all other copies are used only if they ensure that the final copies receives their data in time else they are not used, it follows that the feasible solution computed by DSS_OPT is optimal in execution tile. Summarizing the above discussion, we have the following theorem.

Theorem : The feasible solution calculated by DSS_OPT algorithm is optimal

Numerical example:

We consider here the problem P definite in figure 1 and 2, the DSS-OPT algorithm uses DSS_LWB to compute the earliest possible execution date of all tasks on all possible servers, resulting in the following values b and r (*Table 3)*:

1	b_1	r_1
σ_1	0	15
σ_2	0	10
σ_3	0	20

2	b_2	r_2
σ_1	0	19
σ_2	0	12
σ_3	0	15

3	b_3	r_3
σ_1	14	34
σ_2	10	18
σ_3	16	26

4	b_4	r_4
σ_1	18	23
σ_2	12	18
σ_3	20	29

5	b_5	r_5
σ_1	19	31
σ_2	12	32
σ_3	15	30

6	b_6	r_6
σ_1	22	37
σ_2	18	40
σ_3	24	36

7	b_7	r_7
σ_1	31	41
σ_2	32	52
σ_3	30	55

8	b_8	r_8
σ_1	31	49
σ_2	32	44
σ_3	30	62

9	b_9	r_9
σ_1	49	74
σ_2	49	57
σ_3	53	75

Table 3: *The earliest possible execution date of all tasks on all possible servers for the problem P*

It then computes the smallest makespan of any solution to the P problem :

$$T = \max_{\forall i/\mathrm{SUCC}(i)=\varnothing} \min_{\forall \sigma_r \in \Sigma_i} (r_{i/\sigma_r}) = \min(74,57,75) = 57$$

In our example, the task 9 does not have any successor. The list L_9 of the executions kept for this task in the solution is reduced therefore to the execution $9/\sigma_2$. Thus $L_9 = \{9/\sigma_2\}$. The execution of task 9 on the server σ_2 is scheduled at date 49. Next, The tasks 6, 7 and 8 are the predecessors in G of task 9. For the task 6, the execution $6/\sigma_3$ may satisfy the Generalised Precedence Constraints relative to $9/\sigma_2$. Therefore, this execution is kept and is scheduled at date 24 (b_{6/σ_3}). For task 7, execution $7/\sigma_1$ is kept and is scheduled at date 31…

The table 4 presents the final executions i/σ_r kept by the DSS_OPT(P) algorithm, with their date of execution, in an optimal solution S.

	$1/\sigma_2$	$2/\sigma_1$	$2/\sigma_2$	$3/\sigma_2$	$4/\sigma_2$	$5/\sigma_1$	$5/\sigma_2$	$6/\sigma_3$	$7/\sigma_1$	$8/\sigma_2$	$9/\sigma_2$
b_{i/σ_r}	0	0	0	10	12	19	12	24	31	32	49
r_{i/σ_r}	10	19	12	18	18	31	32	36	41	44	57

***Table 4**: final executions i/σ_r kept by the DSS_OPT(P) algorithm*

We obtain (figure 3) the following optimal scheduling by DSS_OPT(P) algorithm:

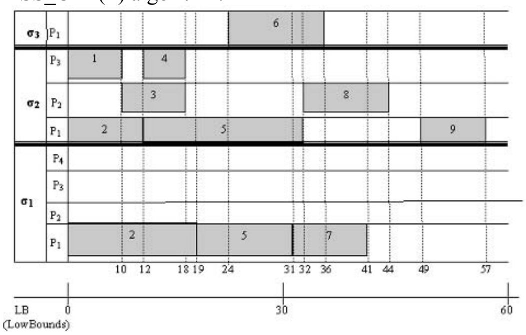


Figure 3 : DSS_OPT algorithm scheduler

3.2 Backup Copies Scheduling (eFRCD Algorithm)

We have just been building a scheduling of the primary copies, with DSS-OPT algorithm. Now, we schedule back-up copies of the tasks with the eFRCD algorithm [16], which allows us to survive failures with the hypothesis [1-failure]. Note that the eFRCD Algorithm was originally proposed to schedule tasks with deadlines in a real-time system with 1-fault hypothesis. We slightly modified it, drooping deadlines, and using earliest date of execution of the tasks and the strong copies[15] definition in ordering list priority. The modified eFRCD algorithm is detailed below.

Some notices and definitions:

- The OL (Ordered List) used by eFRCD to define priorities of tasks will be determined by earliest start execution task's dates by ascending order on all different servers where primary copies are affected with DSS-OPT algorithm scheduling.
- Backup copies are not established for tasks that have already two primary copies with DSS-OPT Algorithm scheduling even it can exist for these tasks the "backup" with good results compared to primary copies.
- **Strong copy definition**: Given a task i, i^P is a strong primary copy, if and only if only a failure of its executing processor may forbid the execution of this primary copy. Because if its executing processor is not in failure, then for each of its predecessors in the DAG whenever the failure is, there is at least one primary or backup copy able to send it its result on time. For example, on figure 3, all primary copies of tasks 1 and 2 are strong copies, because they have no predecessor. The primary copy of task 3 on server σ_2 is strong too because only a failure of σ_2 may forbid its execution. However, the primary copy of task 6 on server σ_3 will not be executed at the scheduled date if there is a failure on σ_2, as it must receive the result of task 3, and there is no other copy of task 3 available. Note that as backup copies of its predecessors are added by eFRCD algorithm, it is possible for an initially non strong primary copy to become strong.

Modified eFRCD Algorithm

Input: Scheduling established by DSS-OPT algorithm
Output: Backup copies scheduling.

Algorithm

```
1.  order the tasks by their earliest start execution as established
          by DSS-OPT algorithm.
          This will generate an Ordered List noted OL;
2.  SL ← Strong-copies in OL;   // SL: Strong-copies List
    OL ← (OL – SL)
    while SL != ϕ do
      for each primary copy i^P of a task i in SL do
        if a backup copy i^B of task i can be established, then
3.        schedule i^B on server σr where it can be scheduled at low
          cost
        end if
      end for
4.    delete from SL tasks whose backup are scheduled
    end while
    while OL != ϕ do
      for each primary copy i^P of task i in OL do
        if a backup copy i^B of task i can be established, then
5.          schedule i^B on server σr where it can be scheduled at
            low cost
        end if
        delete from OL tasks whose backup are scheduled
      end for
    end while
```
End of algorithm

Numerical example:

The execution of our algorithm with the example DAG of figure 2 gives the following results:

1. Ordered List (OL) with precedence constraints by DSS-OPT: {1, 3, 4, 6, 7, 8, 9}; (tasks 2 and 5 are not considered because they have more than one primary copies).

Tasks number	1		3		4		6		7		8		9	
Candidate_servers for backup copies	1	3	1	3	1	3	1	2	2	3	1	3	1	3
Costs on servers	15	20	20	10	5	9	15	22	20	25	18	32	25	22
Low cost Server	1		3		1		1		2		1		3	

Table 5: Table of costs for backup copies on servers

2. Strong primary copies of the list SL = {1, 3, 4, 8};
3. Backup copy scheduling of tasks in list SL:
   - Backup copy scheduling of task 1: server σ_1 , bounds 0 to 15;
   - Backup copy scheduling of task 3: server σ_3, bounds 22 to 32;

- Backup copy scheduling of task 4: server σ_1, bounds from 19 to 24 considering task 2 finish date;
- Backup copy scheduling of task 8: server σ_1, bounds from 31 to 49 considering task5 and backup of task4 finish time.

4. New List OL: {6, 7, and 9}
   - 4.1. Strong primary copies of the list: {7};
   - 4.2. Backup copy scheduling of task 7: server σ_2 , schedule from 31 to 51.
5. New List OL: { 6, 9}
   - Strong primary copies of the list: {};
   - Backup copy scheduling of task 6: server σ_1, schedule from 38 to 53;
   - Backup copy scheduling of task 9: server σ_3, schedule task 9 from max(34+0, 41+12, 49+9) =58, so finish date of 9^B = 58 + 22 = 80.

The next table 6 indicates the Lower Bounds for backup copies for the problem P:

Tasks / Servers	1^B		3^B		4^B		6^B		7^B		8^B		9^B	
	s	f	s	s	s	f	s	f	s	f	s	f	s	f
σ_1	0	15			19	24	38	53			31	49		
σ_2									31	51				
σ_3			22	32									58	80

Table 6: Start dates (s) and finish (f) dates table for back-up copies

Finally, we obtain (figure 4) the following resilient scheduling in the case of one failure :

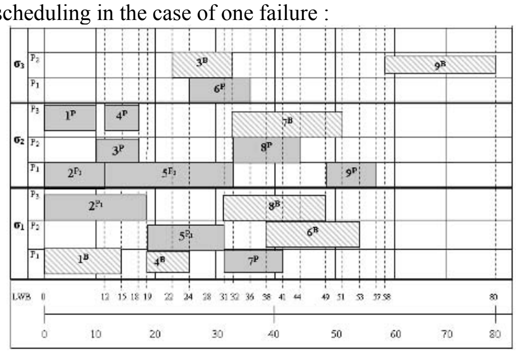


Figure 4 : Modified eFRCD algorithm scheduler

Notes on the figure 4:

- In the last figure σ_1 use 3 processing units, σ_2 use 3 processing units also, but σ_3 use only 2 processing.
- i^P is a primary copy of task i, i^{P1} is the first primary copy of task i, i^{P2} is the second primary copy of task i.
- i^B is the backup copy of task i.

IV. ALGORITHM ANALYSIS

The most computationally intensive part of DSS_OPT(*P*) is the first part DSS_LWB(*P*). In this part, for each task *i*, for each server executing *i*, for each predecessor *j* of *i*, for each server executing *j*, a small computation is done. Thus the complexity of DSS_LWB(P) is $O(n^2s^2)$, where *n* is the number of tasks in *P*, and *s* is the number of servers in DSS. Thus, the complexity of the DSS_OPT(*P*) algorithm is $O(n^2s^2)$. eFRCD is polynomial also, for the reason that eFRCD has two loops (line 3 and line 5), each loop need n*s operations, thus the complexity of eFRCD is $O(n^2s)$. Consequently the global complexity of our new algorithm is $O(n^2s^2)$.

This algorithm has two advantages:

- When there is no fault on servers, our algorithm is optimal as it uses the optimal solution computed by DSS-OPT.
- When there is a fault on one server, our solution is a good one because before the fault the solution by DSS-OPT is better than eFRCD, and after the fault it uses a solution computed by eFRCD, and eFRCD builds a good solution in this case.

Furthermore, note that our solution is guaranteed to finish if one fault occurs, because every tasks has two or more scheduled copies on different servers in the final solution. If more than one fault occur, the solution may still finish, but there is no guaranty now.

The strong copies notion of eFRCD algorithm has been considered in our algorithm because this notion gives an enhanced order of priority. A strong copy is scheduled with a high priority because it is considered as a task without precedence constraints of starting.

We do not establish backup copies for tasks which have already two primary copies with DSS-OPT Algorithm scheduling even it can exist for these tasks the "backup" with good results compared to primary copies. Related to our hypothesis “1-fault”, this option gives scheduling facilities . In our further works, we will study the case of backup copies for tasks, which have more than one primary copy.

The model of failure, as it features at most 1 crash, may seem poor. However, if the probability of any failure is very low, and the probabilities of failures independent, then the probability of two failures will be smaller indeed. Furthermore, the algorithm may be extended to 2 or more failures, by using two or more backup copies.

Finally, the problem solved by this new algorithm suffers from one strong assumption, namely that an unbounded number of tasks may be processed on each server in parallel without any effect on the tasks’ processing time. We mention two cases where this assumption may be satisfied: (a) on systems with enough parallel processors, requiring n processors per server to guarantee schedulability, or (b) constant heavy load on the servers. This limit the usefulness of the direct application of this algorithm to these few cases. However, this algorithm and its model are similar in idea and assumption to the classical Critical-Path Method (or PERT method) [19] as that CPM/PERT method do not consider resources constraints in order to get earliest execution dates. By the way these CPM/PERT results are then used in some real-life systems as the priority values of tasks in some list-scheduling algorithms, the result found by our algorithm may be used as the first step of a list scheduling algorithm, in which the earliest execution dates of primary and backup copies are used as priority values to schedule these copies on the servers of a real-life system.

This page intentionally blank

V. Conclusion

In this paper, we have proposed an efficient scheduling algorithm in which tasks with precedence constraints and communication delays have to be scheduled on an heterogeneous distributed system environment with one fault hypothesis. Our algorithm has combined the DSS-OPT algorithm and the eFRCD algorithm. To provide a fault-tolerant capability, we employed primary and backup copies. But no backup copies were established for tasks which have more than one primary copy.

The result have been a schedule in polynomial time that is optimal when there is no failure, and is a good resilient schedule in the case of one failure. The one fault-tolerant capability was incorporated by using modified eFRCD and the system reliability was further enhanced by using DSS-OPT algorithm where we have optimal scheduling in case of no fault. The execution dates of the primary and backup copies may be used as priority values for list scheduling algorithm in cases of real-life, limited resources, systems.

In our future work, we intend to study the same problem with sub-networks failures. Also, we intend to consider the problem of non permanent failures of servers. Finally, we want to consider the problem of the partial failure of one server, in which one server is not completely down but loses the ability to execute some tasks and keeps the ability to execute at least one other task.

References

[1] J.-Y. Colin, P. Chrétienne (1991). "Scheduling with Small Communication Delays and Task Duplication", Operations Research, vol. 39, n o 4, 680-684, 1991.

[2] M. Maheswaran and H. J. Siegel, "A Dynamic matching and scheduling algorithm for heterogeneous computing systems", *Proceedings of the 7th IEEE Heterogeneous Computing Workshop(HCW '98), pp. 57-69, Orlando, Florida 1998.*

[3] M. Iverson, F. Özgüner, "Dynamic, Competitive Scheduling of Multible DAGs in a Distributes Heterogeneous Environment", *Proceedings of the 7th IEEE Heterogeneous Computing Workshop (HCW '98), pp. 70 – 78, Orlando, Florida 1998.*

[4] H. Topcuoglu, S. Hariri, and M.-Y. Wu., "Task scheduling algorithms for heterogeneous processors". *In8th Heterogeneous Computing Workshop (HCW' 99), pages 3–14, April 1999.*

[5] J.-Y. Colin , M. Nakechbandi, P. Colin, F. Guinand, "Scheduling Tasks with communication Delays on Multi-Levels Clusters", *PDPTA'99 : Parallel and Distributed Techniques and Application, Las Vegas, U.S.A., June 1999.*

[6] A. H. Alhusaini, V. K. Prasanna, C.S. Raghavendra, "A Unified Resource Scheduling Framework for Heterogeneous, Computing Environments", *Proceedings of the 8th IEEE Heterogeneous Computing Workshop, Puerto Rico, 1999, pp.156-166.*

[7] H. Chen, M. Maheswaran, "Distributed Dynamic Scheduling of Composite Tasks on Grid Computing Systems", *Proceedings of the 11th IEEE Heterogeneous Computing Workshop, p. 88b-98b, Fort Lauderdale, 2002.*

[8] M. Nakechbandi, J.-Y. Colin , C. Delaruelle, "Bounding the makespan of best pre-scheduling of task graphs with fixed communication delays and random execution times on a virtual distributed system", *OPODIS02, Reims, December 2002.*

[9] Christoph Ruffner, Pedro José Marrón, Kurt Rothermel, "An Enhanced Application Model for Scheduling in Grid Environments", *TR-2003-01, University of Stuttgart, Institute of Parallel and Distributed Systems (IPVS), 2003.*

[10] P. Palmerini, "On performance of data mining: from algorithms to management systems for data exploration*", PhD. Thesis: TD-2004-2, Universit`a Ca'Foscari di Venezia,2004.*

[11] Srikumar Venugopal, Rajkumar Buyya and Lyle Winton, "A Grid Task Broker for Scheduling Distributed Data-Oriented Applications on Global Grids", *Technical Report, GRIDS-TR-2004-1, Grid Computing and Distributed Systems Laboratory, University of Melbourne, Australia, February 2004.*

[12] M.J. Flynn, "Some computer organization and their e_ectiveness.", *IEEE Transations on Computer, pages 948-960, September 1972.*

[13] Yu-Kwong Kwok, and Ishfaq Ahmad, "Static scheduling algorithms for allocating directed task graphs to multiprocessors", *ACM Computing Surveys (CSUR), 31 (4): 406 - 471, 1999.*

[14] M.R. Garey and D.S. Johnson.,"Computers and Intractability, a Guide to the Theory of NP-Completeness*W. H. Freeman Company, San Francisco, 1979.*

[15] J.-Y. Colin , M. Nakechbandi, P. Colin,"A multi-valued DAG model and an optimal PERT-like Algorithm for the Distribution of Applications on Heterogeneous, Computing Systems", *PDPTA'05, Las Vegas, Nevada, USA, June 27-30, 2005.*

[16] X. Qin and H. Jiang, "A Novel Fault-tolerant Scheduling Algorithm for Precedence Constrained Tasks in Real-Time Heterogeneous Systems" , *Parallel Computing, vol. 32, no. 5-6, pp. 331-356, June 2006.*

[17] B. Randell, "System structure for software fault-tolerance", *IEEE Trans. Software Eng. 1(2) June 1975, 220-232.*

[18] L. Chen, A. Avizienis, "N-version programming: a fault tolerant approach to reliability of software operation", *Proceeding of the IEEE Fault-Tolerant Computing Symposium, 1978 pp. 3-9.*

[19] MR Garey, David S. Johnson: Two-Processor Scheduling with Start-Times and Deadlines. SIAM J. Comput. 6(3): 416-426 (1977).

A Process Family Approach for the reuse of development processes

Maik Thränert and Andrej Werner

Abstract— Development processes are often newly defined for every project. However, a reuse of the process knowledge between different projects rarely takes place. In this paper, we present a concept which permits a general reuse of process knowledge on the basis of the process family approach and as well the project individual customization of processes according to a mass customization.

Index Terms— Organizational management and coordination, Software Engineering Process, Development process reference model, Process model, Mass customization, Process family and Process factory

1 Introduction and Problem Definition

Development projects within those developments are passed through are carried out in many software development enterprises. These development processes are based on known and documented procedures. They are, however, different from project to project, caused by the character of the relative uniqueness of a project.

These development processes can be distinguished in core processes, supporting processes and leadership processes. The real creation of value takes place within the core processes. This creation of value is designed depending on the business, the project, the customers and the product. Supporting processes – e. g. invoicing and reporting – are relatively standardized over the projects. Leadership processes are for control and decision – e. g. in the field of project and multi-project management – and are organized generally for all projects.

The real core processes – as for instance software development, service development or software-service co-design – offer a high potential for the improvement of efficiency. This improvement can be reached during the preparation – in other words, the design of the development process – and the execution of the project. Just like in production processes so-called "operation centre systems" can be used for supporting the development. Such systems allow a process based project control and supervision.

Mass customization is a synthesis of mass production and the production of individual customer products. In this expression, the contrasting concepts of "mass production" and "customization" are combined. With the application of certain technologies and innovative organizational structures, the production of variant-rich and often even customer individual products and services shall be possible at absolutely high product quality and short delivery times to rates of the mass production [1]. The most important aim of the mass customization is the planning and the production of goods and services with that amount of variability, that each customer can find exactly the solution requested by him and the products the he is financially interested in.

In the software sector, reuse and individualization have been playing an important role during the last time. Examples are component development, product line approaches and others. On the other hand, reuse and project-specific customization of process knowledge is inadequately carried out. This leads to a high customization effort during the design of the processes for new projects. The whole development process is frequently completely new developed based on the procedure model, although the projects to be carried out have many common features.

A concept for the adaptation of development processes on the base of individual project features is introduced in this article. The aim is a "mass customization" according to an individual provision of development processes with a high rate of reuse of documented knowledge about development processes. The efficient management of the development process components is therefore required.

First, basic conceptualities are explained in the context of procedure and process models. Then we represent how processes can be defined based on procedure models. The concept of the family based adoption of the procedure models is explained after the exposition of the basic ideas of the system family approach. An example and an outlook will conclude this paper.

2 Basics

2.1 Procedure and Process Models

The explanation of the following concepts is orientated towards the appreciation of the group WI-VM of the German Society of Computer Science, as it is documented e. g. in [2]. Figure 1 shows a summary of the concepts and their relations used here.

The development processes are carried out within the development projects. They are important at the runtime

- *Maik Thränert is with Institute for Computer Science, University of Leipzig, Augustusplatz 10-11, 04109 Leipzig, Germany, E-mail: thraenert@informatik.uni-leipzig.de.*
- *Andrej Werner is with the Information Systems Institute, University of Leipzig, Marschnerstraße 31, 04109 Leipzig, Germany, E-mail: a.werner@wifa.uni-leipzig.de.*

September 30th, 2006.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 309–313.

time of the project. These development processes – and the project itself too – are part of the life cycle of the complete product, e. g. the software life cycle.

The development processes can be represented in a

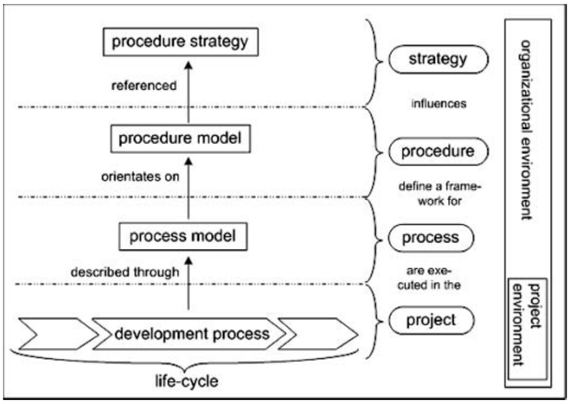


Fig. 1. Basics concepts.

process model. This process model is the result of the process design for the current development project. It depends on the features of this project and is therefore only suitable for this specific project. The process model is consequently also part of the project plan.

A procedure model is the reference model for the development processes. It describes in an abstract way in which states the system to develop can be. A procedure model is the base for the process model, which is formed depending on the specific project features.

The procedure model is influenced by the basic procedure strategies. Such a procedure strategy describes a vision or an idea about the development of the application [3, p. 33]. Version oriented or risk oriented approach are examples of procedure strategies.

Variants for the derivation of process models

To derive process models from procedure models, different ways are conceivable as represented in the following.

Variant 1: A generic procedure model from the literature which seems to fit to the project environment is adapted to the conditions of the organisation and the concrete project requirements. A specific procedure model is designed for the organisation and the projects on that base.

This specific procedure model is then refined to a project-specific process model. This procedure reoccurs correspondingly at every project. It is obvious that the reuse is carried out only on the high abstraction level of the generic procedure model. The same or similar customizations are performed repeatedly, see figure 2.

Variant 2: A reference procedure model, suitable for the

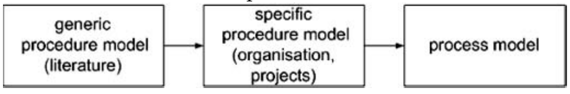


Fig. 2. Variant 1.

organisation and the project type and based on a generic procedure model, is selected. This reference model is adapted to organization-specific and project-specific conditions. If a reference process is also part of the reference model, then this can be adapted for the concrete project (customization). If not, a process model is made based on the adapted reference model. The reference model (procedure and/or process model) can be extended by organization-specific or project-specific attributes and therefore serve as a template for further projects.

The level of reuse in this variant is higher than in the first one, because specific features are already documented in the reference models, see figure 3.

Variant 3: Here the reference procedure model is the

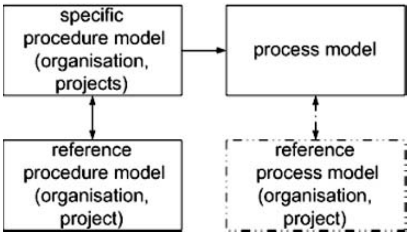


Fig. 3. Variant 2.

foundation for the adaptation. This reference procedure model must be suited for the organisation and project-specific conditions. If a reference process is available, then this is adapted correspondingly (customization). If not, the reference procedure model must be used as starting point of the customizations. This way is the same compared to Variant 2.

The project features which influence the customization of the process model are now identified. The common and different features are documented in a feature model. With every project, the knowledge therefore increases. The probability increases that certain development elements (so called process components which are different from the reference model) can be applied for further development processes. It is prerequisite that features of a process component are related to features of a project.

With this approach, the development process does not have to be described again and again from scratch. Common and varying process components are reused: a configuration of the process model. The result is a development process which was configured on the basis of already existing process components. Process components that do not have a fitting template till now are created and added to the knowledge base. So they are able to be used again in the future.

The highest reuse of process components of a procedure model is achieved with the variant 3. The costs for the individualization of a development process are the lowest. By reusing of the process knowledge, the quality and the maturity also increase. This aggregated knowledge can also be changed.

This variant is introduced in detail in chapter 3. Basic ideas of system families are explained in the following part.

2.2 System Families

The domain engineering was inspired by the thought of the effective reuse of software in a domain, particularly of identifying features in a domain [4, 5, 6, 7, 8, 9 et al.]. Starting out from the objectives of the stakeholder interest, the requirement analysis describes the common and different identifying features of the examining domain. This represents the basis for a system family and influences particularly the comprehensibility, maintainability, usability and reusability of a system or a system family [10].

A domain is the knowledge or application field which is defined by the stakeholders. A system family is characterized according to [7] as a set of systems which have many common characteristics building on a set of common components. The aim of the domain development is to be able to develop new systems or new system variants of a high quality in a fast and economical way using a high degree of reuse of important assets. Based on the domain development, the development of the application starts at the point where a concrete variant of the system family is viewed and developed. These experiences in the course of an iterative development flow into the further development of the system family.

The differences between systems of a system family arise from different requirements on single system expressions. The requirements on the system can be interpreted at a high abstraction level as features of the systems. The features which have all systems of the system family define the commonality and the features which vary from system to system, describe the variability of the system family. The feature modelling represents these aspects in a model.

The concept of the feature modelling was developed in the context of FODA [11] and extended by further work, for example of [7, p.82pp], extended. The feature models consist of one or several feature diagrams and supplementary information. The feature diagrams serve as a graphic representation of the features in the form of structure trees. Textual composition rules serve the description of crossways relations which are not viewable in a feature diagram. A feature in FODA represents a clearly visible quality or characteristics of a system [11, p.3] to the user. One of the basic compromises that a system architect has to solve is the decision of the time, at which a feature is selected or when it should get its value —in other words when a feature is "tied". In FODA the concept “binding time” for a feature was introduced and three times, compile-time, load-time or run-time [11], were described.

Features are classified in four different categories according to [12, p.3]. External features are not direct components of the system but the platform. The necessary features represent the things in common of the systems and must be tied in every case. Optional features, however, cannot be tied and increase the functionality of a system at their integration. The alternative features serve for the choice of a feature expression from a group of features as the last. The selection criterion which makes the choice possible between the alternative features is described as a variation point.

In the context of the RSEB method, Jacobson et al. introduce the idea of the variation point in [13, p.98]. It serves as a representation of the places in which differences can appear in a group of similar systems. In the feature modelling, a variation point serves as the representation of the choice of alternative or optional features. While the feature models are used for the implementation independent representation of systems, variation points make the consideration of variability possible (from the requirement analysis to the design and the implementation) in every stage of the development. Appearing of variability in features, in processes, on the data scale, in the data format, in the access to system, in user interfaces, in system interfaces and in the quality identified Bühne et al. in [14, p.14-16]. Variability in processes means, a functionality of two products can be realized by activities by an order differing from each other.

As mentioned, the concept of the system families represents an opportunity of the reuse of software assets. These assets consist primarily as components source code or the know-how for the production of software artifacts (documentation requirements etc.). The development also represents a significant competition factor. Sutton et. al. noticed in [15] that procedures have influence on the software product development and concrete developments are derived from them. If the development within this procedure varies, the result of the development and vice versa also vary. Furthermore, [15] noticed that procedures do not stipulate any rigid development for software products but rather represent an action framework in which the individual development occurs. These individual developments vary dependending of different criteria (project type, product, staff, resources etc.).

Sutton et. al. understand a concrete development process as a member of a process family which houses a variety of options with respect to the mentioned criteria. On the base of these ideas and the work on the Process Factory [16], a concept for the individualization of process models on the base of process families is introduced in the following chapter.

3 Family Approach for Process Models

To be able to use the introduced family approach for process models, several activities are required, see *figure* 4.

At first, process model families have to be developed, what corresponds to the concept of domain engineering. As a result, the features are documented and stored in the process model family repository in the form of feature models and configuration models. The starting point for this domain engineering is either an analysis of former projects or the know-how of the so called “domain engineer”.

The development of the individual process models is done based on the process model family repositories. Therefore, the features of the project have to be analyzed (analysis). Then the process family configured (configuration), before the suitable process model can be produced (transformation). These activities are part of the configuration period of the project.

During the analysis, the project environment is examined, and a suitable process family with respect to procedures and strategy is selected, see figure 1. This process is described as scoping. An analysis of the project due to defined project features and the documentation are then carried out in form of a project-specific feature model. As a result, the choice of a suitable process family and the characterization of the project requirements by the feature model arise.

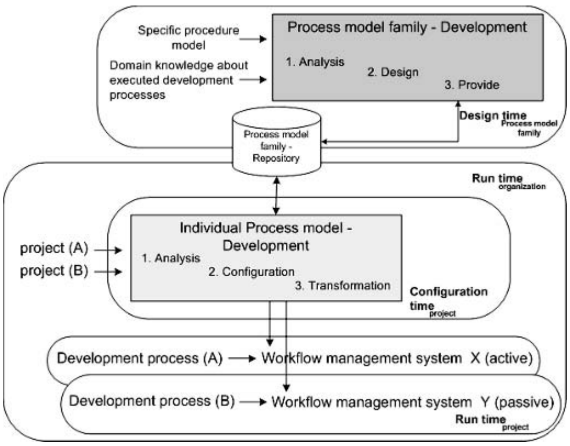


Fig. 4. Reuse of development knowledge by process families.

During the configuration, the comparison of the project feature model with the features of the process family is carried out. Process components are identified, which are used in the concrete development process. Furthermore, process components can be parameterized with the help of features. Process components, which are till now not available and required in the process, have to be created. The result of the configuration is a concrete individual development process, expressed in the language of the process model.

The following transformation makes the construction of a concrete, individual development process model in the language of the workflow management system[1] possible. This allows an instantiation of the process model and so the execution of the project. The development process model resulting from the configuration, descriptions of the target language as well as transformation rules and specific attributes of the project (resources, names, addresses etc.) are required for input.

The development becomes suitably deployed and executed if supported by the workflow management system. The process is executed, what means that the operation centre system steers actively the project with resources, employees, documents etc. The states of the activities are recorded by the workflow management system itself. In the case of a passive workflow management system, the development processes are used merely for the documentation of the project. The project manager is responsible for controlling the project. States of the activities in the development process are recorded and documented by the employees. Figure *5* summarizes the given explanations.

[1] In German we mean an operation system called "process operation system" (Prozessleitstand) like a "production operation system" (Produktionsleitstand).

4 USE-CASE

A CRM (customer relationship management) service provider shall serve as an example for the introduced concept. The core business of this provider is the launch and

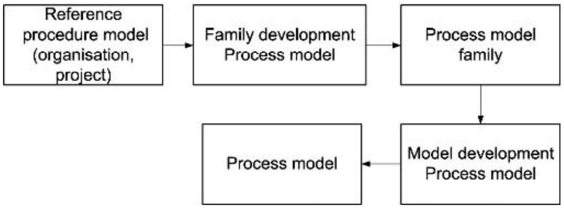


Fig. 5. Family approach for process models.

the customization of CRM application software for customers in the area of mechanical engineering. This focus on a specific customer industry has an advantage: From the perspective of the service provider, projects of this type behave relatively similarly to each other. Through the huge amount of such projects, it is vital for the service provider to use a project-specific adjustment of development processes.

According to the way described in chapter 3, the development of a process model family was the first step, see [17, p. 28]. Parts of this are collection of features and variation points of the process model family and its documentation in a feature model. Such a feature model is illustrated in figure 6 for two selected features.

First, the type of the project has to be distinguished. A project can be whether a project that introduces a CRM solution for a customer (introduction project), a project that extends or modifies the functionality of an existing CRM solution (extension or change project). Blueprint and consulting projects can be carried out in addition to the other

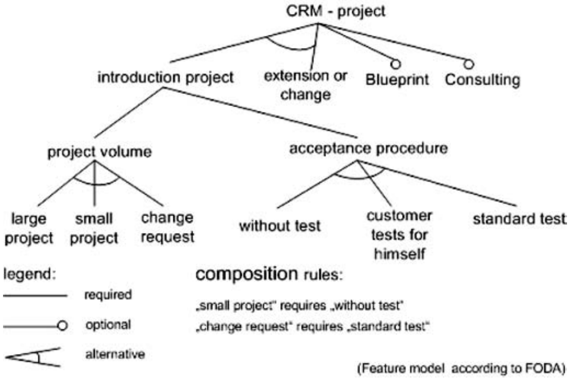


Fig. 6. An project-spezific example for a feature model.

types of projects. To keep the model in figure 6 compactly, we focus only on introduction projects.

Project's volumes and the acceptance procedure at its end play an important role for the way the introduction project goes. If it is a "small project", project preparation and project acceptance are thinner compared to "large projects". The internal release is completely discontinued at

This page intentionally blank

"small projects" and "change request". Depending on the concrete attribution of this feature in the project, more or less resources have to be included in the project preparation and the acceptance phase. The whole activity "internal release" could possibly be dropped.

The feature "acceptance procedure" defines how the test of the results at the end of the project shall be carried out. It is possible that: 1.) there is no test by the service provider at all – the result of the project is accepted by the customer as it is; 2.) the customer tests for himself and a list of open points is given to the service provider as a result of the test or 3.) customers and service provider perform the test together.

Depending on the attribution of the feature "acceptance procedure", the design of the phase "acceptance procedure" changes in the procedure model. The roles of service provider and customer have to be assigned to the related activities or the phase has to be left out completely.

To be able to make a concrete process model for the project out of the process family model made above, at first the attributes of the concrete project have to be analyzed and documented within the feature model. These attributes influence the concrete process model on the defined variation point in the way described above.

The shown approach can be used appropriately in the introduced domain. This is due to the characteristics of the domain: The projects are very similar to each other, and the attributes that determine the process are well-known and clearly defined.

5 Review and Outlook

The paper could merely introduce the basic ideas of the concept. The following work still has to be done:

- Extensions of feature models and configuration models: It is very difficult to build up these models since a deep specialized knowledge about project processes and the interdependence between the process and the features of the project is required. Furthermore, the problem is that the corresponding employees have to be willing to explicate their high-quality knowledge.

- Granularity of the reuse: It has to be verified at which granularity process constituents have to be defined for a sensible reuse. If the elements are too rough, the arising process models are not adequate for the project. If they are too fine, the effort for the feature modelling and the definition of the interdependencies strongly arises.

- Practical relevance: The concept has to be detailed and tested on the base of suitable use cases. Such suitable use cases have to fulfil the following criterion to be useful: Their users have to work on sufficiently similar projects so that a sensible modelling and a sensible use of the concept can be reached. This is given in the use case introduced in chapter 4. Nonetheless, broader practical experiences must be gained to prove the usefulness of the concept.

The concept of the reuse of process knowledge introduced in this paper tries to realize a reuse during the transformation of the process model into the real development process, see figure 1. On the other hand, a reuse at a higher level of abstraction – meaning at the level of the procedure model – is possible. In this case, the features of procedure model families have to be analyzed and its components could be reused suitably. Furthermore, it is possible to use the concept at two levels at the same time.

References

[1] Pine, B., J. Massgeschneiderte Massenfertigung: neue Dimensionen im Wettbewerb. Wirtschaftsverl. Ueberreuter, Wien 1994.

[2] Ralf Kneuper, Günther Müller-Luschnat, Andreas Oberweis (Hrsg.): Vorgehensmodelle für die betriebliche Anwendungsentwicklung, B.G. Teubner, Stuttgart/Leipzig, 1998.

[3] Bremer, G.: Genealogie von Entwicklungsschemata. In: Kneuper R. et. al.: Vorgehensmodelle für die betriebliche Anwendungsentwicklung. Teubner-Verlag, 1998.

[4] Prieto-Díaz, R. Domain Analysis: An Introduction. Software Engineering Notes, 15(2), 1990, S. 47-54.

[5] Wartik, S., Prieto-Diaz, R. Criteria for Comparing Reuse-Oriented Domain Analysis Approaches. International Journal of Software Engineering and Knowledge Engineering, 2(3), 1992, S. 403-431.

[6] Arango, G. Domain Analysis Methods. In Software Reusability, Schäfer, R. Prieto-Díaz, Matsumoto, M. (Eds.), Ellis Horwood 1994.

[7] Czarnecki, K., Eisenecker, U. Generative Programming: Methods, Tools and Applications. Addison-Wesley, Boston 2000.

[8] Cleaveland, C. Program Generators using Java and XML. Prentice Hall, New Jersey 2000.

[9] Clements, P., Northrop, L. Software Product Lines: Practices and Patterns. Addison-Wesley, Boston 2002.

[10] Foreman, J. Product Line Based Software Development- Significant Results, Future Challenges. Software Technology Conference, Salt Lake City 1996.

[11] Kang, K., Cohen, S., Hess, J., Novak, W., Peterson, S. Feature-Oriented Domain Analysis (FODA) Feasibility Study. CMU/SEI-90-TR-21. Software Engineering Institute, Carnegie Mellon University, Pittsburgh 1990.

[12] Gurp, J., Bosch, J., Svahnberg, M. Managing Variability in Software Product Lines. 2000, http://www.jillesvangurp.com/publications/ managingvariability-inSPLs.pdf, abgerufen am 10.10.2004.

[13] Jacobson, I., Griss, M., Jonsson, P. Software Reuse - Architecture, Process and Organization for Business Success. Addison-Wesley, 1997.

[14] Bühne, S., Halmans, G., Lauenroth, K., Pohl, K. Variabilität in Software-Produktlinien. In: Böckle, G., Knauber, P., Pohl, K., Schmid, K. (Hrsg.) Software-Produktlinien Methoden, Einführung und Praxis. dpunkt-Verlag, Heidelberg 2004.

[15] Sutton, S.M., Osterweil, L.J. Product Families and Process Families. Proceedings of the 10th International Software Process Workshop (ISPW '96), IEEE (2)97, S.109-111.

[16] Werner, A. Fertigung kundenindividueller Geschäftsprozesse mittels einer Process Factory. Lectures Notes in Informatics (LNI) - Proceedings GI-Informatik 2005, Volume P-67, 2005, S. 64-68.

[17] Peter Wetzel, Alexander Soutchilin: Integration Engineering im Maschinen- und Anlagenbau. In: Fähnrich, Klaus-Peter et. al.: Integration Engineering – Umsetzung von kooperativen Geschäftsprozessen auf eine internetbasierte IT-Struktur. Universität Leipzig, 2005.

Organizational Change Measurement via Change Metrics

M. S. Camara, L. Kermad, A. El Mhamedi
MGSI « Modélisation et Génie des Systèmes Industriels »
IUT de Montreuil- Université de Paris 8. 140, rue de la Nouvelle France
93100 Montreuil Cedex,France E-mail : mamadousamba_c@yahoo.fr

Abstract – Business Process Reengineering (BPR) has been popularized in recent years as the most important technique for restructuring business operations to achieve dramatic improvements in profitability and sustainable competitive advantage [3]. The Re-engineering activity is a transformational change, moving the business outside its current «rules and games » [5]. Change management is then necessary to manage people trough the emotional ups and downs lead to the massive change and then prevent resistance. Several process evaluation approaches exist to analyze the impact of business processes structural complexity on their performance. To achieve this objective structural and operational metrics are defined upon processes. These metrics are mostly adaptations of software Process or product Metrics. However, these approaches do not target performance problems lead to organizational and business change, that inevitably occur in projects like BPR. This paper defines change metrics that measure the change operated on business process models. Our metrics are also inspired by change metrics defined in Software engineering [6]-[7]-[8]. This work is a first step towards a quantitative and predictive change management methodology to prevent risk related to organizational change in Reengineering projects.

I. INTRODUCTION

In relevant literature of risk management in software engineering, we can find predictive risk management methods based on decision tree modeling [7]. These methods allow fault detection in software modules earlier in the software development process. However, the literature on the risk management in BPR project does not give substantial and reliable quantitative methodologies to manage risk related to organizational change. BPR projects, or other projects that require a BPR like ERP implementation, suffer from low rate of success. BPR projects are characterized by the occurrences of problems, which may lead to major restructuring of the business processes. BPR requires changes to organization structure, roles, job design, and material/information flow [5]. Looking at the research done from the perspective of business people, we can find a variety of proposals for the evaluation of processes. There are mostly from the point of view of the results obtained in the execution of these processes [1], with the use of operational metrics. This evaluation of performance and results of business processes include aspects such as time and cost of the process [2]. There is also a literature related to the evaluation of structural complexity of business processes using structural metrics. Its objective is to establish relation between operation and structural metrics and then proof the influence of the process structural complexity on its performance. These approaches address performance problems lead to process complexity but do not address those lead to organizational and business change, that inevitably occur in projects like BPR. This paper defines structural and operational change metrics that measure the change operated on business process and working environment models. Our metrics are also inspired by change metrics defined in Software engineering [6]-[7]-[8]. This work is a first step towards a quantitative and predictive methodology, like in software engineering, to prevent risk related to organizational change earlier in Reengineering projects.

II. EVALUATION MEASURES FOR BUSINESS PROCESS MODEL

In Elvira's work [2] business process metrics fall into two major categories: operational and structural [2]-[3]. Operational metrics are those that measure how the process is performing through time [3]. While operational metrics deal directly with the dynamic properties of business processes, structural metrics deal directly with static properties [3]. Structural metrics [1]-[2] has been grouped into two main categories: Base measures and derived measures. Base measures are calculated by counting the different kind of elements that a business process model is composed of represented in BPMN (Business Process Modeling Notation) [1]. These static, or structural, properties strongly influence the performance of the process. A benchmark is also realized upon business processes by collecting measurements against both operation and structural metrics. The collected measurements can help to analyze the impact of deficiencies in the process structure on its performance problems. The structural metrics represent a quantification of business process structural complexity. They are composed of 43 Base measures and 14 derived measures.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 315–319.

A. Bases measures

Base measures are simple metrics calculated using the Business process model with BPMN. These base measures consist principally of counting the business process model's significant elements. Bases measures are distributed according four categories of elements [2]:

- 37 in Flow Objects category
- 2 in Connecting Object category
- 2 in swimlane category
- 2 in Artifacts category

In the Flow Objects category, base measures are also grouped according to the common elements to which they correspond [2]

- o 23 for Event element
- o 9 for Activity element
- o 5 for Gateway element.

Somme of the bases measures [1] are shown in TABLE I.

B. Derived measures

Derived measures are more complex metrics resulting of aggregation of bases measures. Derived measures allow seeing proportions that exist between the different elements of the model. TABLE II show some derived measures defined by Elvira [1]. The formulas that require base measures that are not presented in TABLE I are omitted. The complete tables corresponding to base and derived measures are available in [1].

TABLE I
BASE MEASURES BASED ON BPMN

category	element	Core element	Metric Name	Base measure
Flow Objects	Event	Start Event	NSNE	Number of Start None Events
			NSTE	Number of Start Timer Events
		Intermediate Event	NIMsE	Number of Intermediate Message Events
			NIEE	Number of Intermediate Error Events
		End Event	NECaE	Number of End Cancel Event
			NELE	Number of End Link Event
	Activity	Task	NT	Number of Task
			NTC	Number of Task Compensation
		Collapsed Sub-process	NCS	Number of Collapsed Sub-process
			NCSA	Number of Collapsed Sub-process Ad-Hoc
	Gateway	Inclusive (OR)	NID	Number of Inclusive Decision/Merge
		Parallel (AND)	NPF	Number of Parallel Fork/Join
Connecting object		Sequence Flow	NSFA	Number of Sequence Flow between Activities
			NSFE	Number of Sequence Flow Incoming from Events
		Message Flow	NMF	Number Of Message Flow Between participants
Swimlanes		Pool	NP	Number of Pools
		Lanes	NL	Number of Lanes
Artifacts		Data Objects (Input)	NDOIn	Number of Data Objects-In
		Data Objects (Output)	NDOOut	Number of Data Objects-Out

TABLE II
DERIVED MEASURES BASED ON BPMN

Name	Formula	Metric
TNSE	…	Total Number of Start Events of the Model
TNIE	…	Total Number of Intermediate Events
TNEE	…	Total Number of End Events
TNE	TNE =TNSE +TNIE +TNEE	Total Number of Event
TNT	…	Total Number of Task of the Model
TNDO	TNDO = NDOIn + NDOOut	Total Number of Data Objects in the Process Model
PDOPIn	PDOPIn = NDOIn / TNDO	Proportion of Data Objects, as Incoming Product and the total of Data Objects
CLP	CLP = NMF / NP	Connectivity Level Between Pools
PLT	PLT = NL / TNT	Proportion of Pools and/or Lanes of the Process and Activities in the Model

C. Operational Metrics

Operational metrics tend to be meaningful only if measurements are made and recorded periodically for comparison [3]. We can find in the literature the following operational metrics:

- Units produced per day[3], Percent manufactured correctly[3]
- Process Mean time [2]-[3], Processing time [9], Cycle time [9].
- Process cost [2]-[10]

The business process complexity evaluation approach is a good starting point for managing performance problems related to business process change. The group of metrics of the process complexity evaluation has to be adapted in order to measure, not the complexity of a process, but the change between a business process and its restructured version after BPR.

II. CHANGE METRICS IN SOFTWARE ENGINEERING

Demeyer [6] say that to really understand evolving software, the changes themselves are the critical factor. Gokhale [7] considers that there is a one-to-one correspondence between the number of changes made to the executable code of the software module and the number of faults found in this one. Change metrics are defined to measure this change starting from CRs (Change Reports) [7] which documents the changes carried out on the code of the modules. Demeyer [6] defines change metrics by comparison between successive versions of object-oriented software systems source code. He defines first metrics, which we can call base measures, grouped in three categories of [6]:

- Method Size (computed for each method)
    - Number of message sends in method body.
    - Number of statements in method body.
    - Lines of code in method body.
- Class Size (computed for each class)
    - Number of methods in class.
    - Number of instance variables
    - Number of class variables
- Inheritance (computed for each class)
    - Number of immediate children of class.
    - Number of inherited methods
    - Number of overridden methods,

Global change metrics, which we consider as derived metrics, were defined for software systems starting from metrics defined above [6]. The following derived change metrics show changes related to classes in the software system [6]:

- Classes **Ini**: Number of classes in the Initial version
- Classes **Rmv** : Number of Removed classes during the transition between two versions
- Classes **Add** : Number Added classes during the transition between two versions
- Classes **Ret** : Number Retained classes during the transition between two versions
- Classes **Fnl** : Number of classes in the Final version

III. ADAPTATION OF THE BUSINESS PROCESS EVALUATION METRICS

Structural metrics of the process complexity evaluation approach have to be adapted to capture change on processes. The new created metrics are called base and derived change metrics. Operational change metrics are inspired of process evaluation operational metrics. They should represent performance variation during the business process reengineering. The definition and calculation of this metrics will require two version of the same business process:

- Business Process Model Before BPR **(PM_B_BPR)**

- Business Process Model Before BPR **(PM_A_BPR)**

A. Base Change Metrics

As Demeyer [6], we define our change metrics starting from metrics presented in section II. Every Base Measure is replaced by 3 new Change Base Measures**.** These Change Base Measures are calculated by comparing the initial version of the business process model and its restructured version. This comparison determinate how many units are added removed and retained for every kind of element in the model. Examples for Sequence Flow between Activities, Task and Lanes elements are shown in TABLE II. We should have at all 129 (3 * 43-the 43 base measures defined by Elvira [1]-). Contrarily to Demeyer [6] we have just count Removed, Added and Retained element from which we can deduce Initial and Final number of elements. We want to have the minimum number of change base metrics, just those needed to calculate derived change metrics.

B. Derived Change Metrics

Derived change metrics, see TABLE IV, are inspired of process complexity evaluation derived metrics presented in TABLE II. Their calculation formulas are based on base change metrics combinations. For the derived change metrics that include proportion calculation, the division is done with the total element in the process model before BPR. We consider that the size of the initial process is the reference for measuring the change in proportions. This size is given by addicting the number of retained elements and the number of removed one (**Ret** + **Rmv**) - see TABLE IV.

TABLE III
CHANGE BASE METRICS

CBM	CBM	Description
NT	NAddT	Number of Added Task (Number of Task In both models)
	NRmvT	Number of Removed Task (Number of Task Only in model PM_B_BPR)
	NRetT	Number of Retained Task (Number of Task Only in model PM_A_BBR)
NL	NAddL	Number of Added Lanes
	NRmvL	Number of Removed Lanes
	NRetL	Number of Retained Lanes
NSFA	NAddSFA	Number of Added Sequence Flow between Activities
	NRmvSFA	Number of Removed Sequence Flow between Activities
	NRetSFA	Number of Retained Sequence Flow between Activities

TABLE IV
DERIVED CHANGE METRICS

Name	Formula	Metric
TNAddSE	…	Total Number of Added Start Events
TNAddIE	…	Total Number of Added Intermediate Events
TNAddEE	…	Total Number of Added End Events
TNAddE	TNAddE =TNAddSE + TNAddIE + TNAddEE	Total Number of Added Event
TNRmvT	…	Total Number of Removed Task of the Model
TNRetDO	TNRetDO = NRetDOIn + NRetDOOut	Total Number of Retained Data Objects in the Process Model
PAddDOPIn	PAddDOPIn = NAddDOIn / (TN**Ret**DO + TN**Rmv**DO)	Proportion of Added Data Objects, as Incoming Product and the total of Data Objects in the PM_B_BPR
CLPAdd	CLPAdd = NAddMF / (N**Ret**P + N**Rmv**P)	Connectivity Level Between Pools Added
PRmvLT	PRmvLT = NRmvL / (TN**Ret**T + TN**Rmv**T)	Proportion of Removed Pools and/or Lanes of the Process and Activities in the PM_B_BPR

This page intentionally blank

C. Operational Change Metrics

Operational Metrics presented is section II-C should be calculated first for the process model before and after BPR. The comparison of these two values gives the operational change metrics as cost, time and Percent manufactured correctly changes for the processes. For the predictive management of performance problems lead to change we propose the use of Bayesian network structural learning. Operational change metrics will be considered as the Bayesian network's dependant or response variables. Predictor or independent variables will be the bases and derived change metrics. The Bayesian network formalism is becoming increasingly popular in many areas such as decision aid, diagnosis and complex systems control, in particular thanks to its inference capabilities, even when data are incomplete. The role of Bayesian network will be to establish influence of business process changes, quantified by change metrics, and the increasing or decreasing of process results, quantified in operational change metrics.

IV. CONCLUSION AND DIRECTIONS FOR FUTURE RESEACH

In summary we have defined change metrics to answer the lack business process evaluation methods addressing performance problems lead to change. In one hand we were inspired by change metrics defined software systems [6]-[7]-[8] in the way of capturing the change on business processes. In the other hand, our base and derived change measures and our operational change metrics are adaptations and extensions of structural complexity and operational metrics defined in process evaluation [1]-[2]. In future research we will address change metrics validation in real case, and their use in predictive business change management methodology.

REFERENCES

[1] Elvira R. Applying Software Metrics to evaluate Business Process Models CLEI electronic journal Volume 9, 2006

[2] Elvira, Evaluation Measures for Business Process Model, Proceedings of the Symposium of Applied Computing Dijon France 2006

[3] Gary S. Tjaden Business Process Structural Analysis, Georgia Tech Center for Enterprise Systems, june 2001

[4] M. S. Camara, L. Kermad, A. El Mhamedi Quantitative Risk Analysis in ERP Projects Based On The Change Measurement, Proceedings 12 th IFAC Symposium on Information Control Problems in Manufacturing

[5] Mike Dale, The Re-engineering Route to Business Transformation, The BPR Handbook November 1994.

[6] Serge Demeyer, Finding Refactorings via Change Metrics, Proceedings of the 15 th ACM SIGPLAN conference on Object -Oriented Programming, systems, languages, and applications.

[7] Gokhale S. and. Lyu M.R, "Régression Tree Modeling for the Prediction of Software Quality", *In Proc. of ISSAT'97,* Anaheim, CA, March 1997

[8] Kaszycki G., Using Process Metrics to Enhance Software Fault Prediction Models, *FastAbstract ISSRE* Copyright 1999

[9] N Kock Changing the focus of Business process redisgn from activity flow to information flow

[10] A W Scheer ARIS- Business Process Modelling, Thrid Edition, Springer 1999

Unicode Searching Algorithm Using Multilevel Binary Tree Applied on Bangla Unicode

Md. Akhtaruzzaman
Department of Computer Science and Engineering, International Islamic University Chittagong Dhaka Campus
akhter.mail@gmail.com, akhter900@yahoo.co.uk

Abstract- **Unicode Searching Algorithm using multilevel binary tree is proposed to search the Unicode in efficient way. The algorithm is applied on Bangla Unicode searching to convert Bijoy string into Unicode string. Firs, the algorithm build a multilevel binary tree based on ACII code with its corresponding Unicode. The tree is build from a multilevel binary sorted data containing ASCII code and its corresponding Unicode. The data must be sorted based on ASCII code. The algorithm takes Bangla Bijoy string as input value and output the same string in Unicode format. The input Bijoy string must be in Unicode readable format**

I. INTRODUCTION

Fundamentally, computers just deal with numbers. They store letters and other characters by assigning a number for each one. Before Unicode was invented, there were hundreds of different encoding systems for assigning these numbers. These encoding systems also conflict with one another. That is, two encodings can use the same number for two different characters, or use different numbers for the same character. Any given computer (especially servers) needs to support many different encodings; yet whenever data is passed between different encodings or platforms, that data always runs the risk of corruption. Unicode is changing these all. Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language.

Bangla is the mother tongue of Bangladeshi people and the second most popular language in India. It is a reach language and more than 10% people speak in Bangla in the world. But in field of Computer Science the research on this language is not good enough. There exists some Bangla writing software like Bijoy, Avro, Akkhor etc. Bijoy is the most popular and oldest software which is used to write the plain test only. Avro uses Unicode to write Bangla sentences. Now a day Unicode format is used to write any language and sometimes it is necessary to convert the plain text into Unicode format. So searching Unicode is necessary. In Unicode, there are 65,535 distinct characters that cover all modern languages of the world. So Unicode searching must be efficient and reliable for all languages. In some Unicode searching algorithm, especially in Bangla, there exist only 'if – else' condition. Some one uses the 'Hash Table' to develop Unicode searching method. Here a Multi Level Tree based Unicode searching algorithm is proposed which will be more efficient for Unicode searching on different languages.

II. PRELIMINARY STUDIES

A. Bijoy String to Unicode Readable Format String

There exist 11 independent characters (vowel) and 39 dependent characters (consonant) [6] in Bangla literature. There also exist some independent and dependent character symbols called 'Kar' and 'Fala' respectively. These symbols must be used with a character. A large number of Complex Characters (combination of two of more characters) exist in Bangle language. A single Complex Character may contain two or more independent or dependent characters, must joined with a symbol named 'Hasanta' (্).

In plain text all the characters and symbols may placed independently anywhere in a sentence. Bijoy follows this rule. But Unicode maintains a unique format to use the symbols with a character. In Unicode the symbols must be used after a character with no gap between them i.e. "character + symbol". But in some cases like 'Chandrabindu' or 'Ref' the placement is different. 'Chandrabindu' must be used after a character if no symbols are exists with that character i.e. "character + Chandrabindu". If any symbol is exist with a character then the 'Chandrabindu' is used after the symbol i.e. "character + symbol + Chandrabindu". 'Ref' must be placed before character.

Figure 1 shows some examples, representing Bijoy plain text and its Unicode readable format of Bangla sentences. Figure 1(a) contains a complex character 'ম্ম' that forms with two ম s, Bangla dependent characters (consonant), i.e. ম + ্ + ম = ম্ম .

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 321–326.

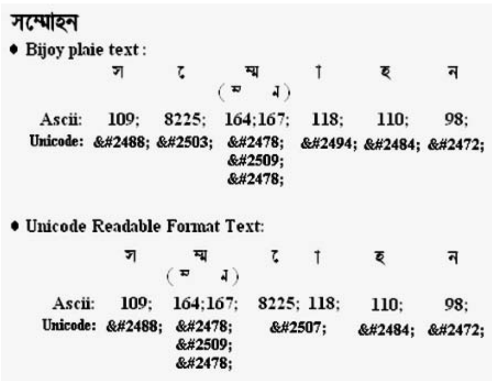


Fig. 1 (a)

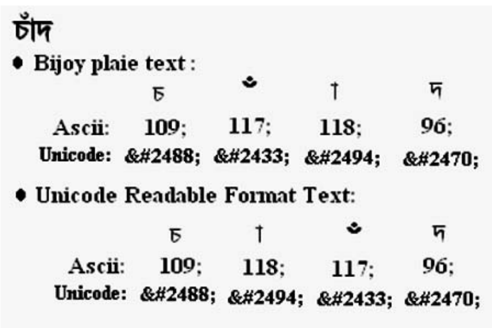


Fig. 1 (b)

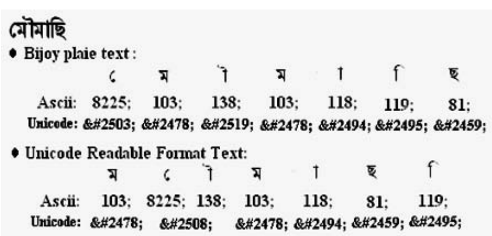


Fig. 1 (c)

Fig. 1. Bijoy plain text and its Unicode readable formatted text showing ASCII and Unicode of corresponding character.

B. Multi Level Binary Sort

Binary search is a well known and efficient search algorithm in the field of Computer Science. To apply this algorithm on a list it must be sorted in increasing or decreasing order. There exist a lot of sorting algorithms which are used to sort data in non-increasing or non-decreasing order. Here the term 'Binary Sort' is used because the method follows the technique of Binary Search algorithm and rearrange the data into a in order tree format where its first value indicates the middle value of the data list, second value indicates the middle value of the first dividing part of the previous data list and so on.

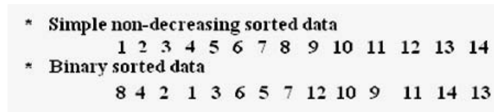


Fig. 2. Binary Sorted data generated from a simple non-decreasing sorted list.

Figure 2 shows the Binary Sorted data generated from a non-decreasing sorted list. Here in Binary sorted data the first value '8' indicates the middle value of Simple sorted data. Second value '4' is the middle value of the first division (1 to 7) and third value '2' is the middle value of the first division (1 to 3) of the previous first division of Simple sorted data.

Here another term 'Multi Level' is used with the term 'Binary Sort' because this method is applied on multilevel sorted data. Figure 3 shows the Multi Level Binary Sorted Data formation.

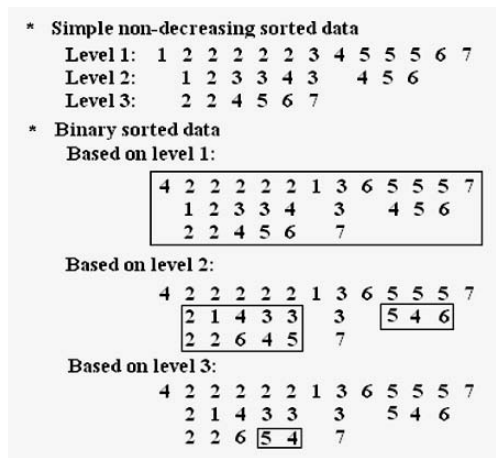


Fig. 3. Three level Binary Sorted data generation from three level non-decreasing sorted data.

Here in the first level of the Simple non-decreasing data has 5 twos and 3 fives which are treated as single integer each. So applying Binary Sort on the basis of first level the sequence of 2 and 5 will remain unchanged. Now come to the second level. Here exist 2 threes for common value '2' of the first level and is treated as single three. Applying Binary Sort on the values of common two and five of first level the values will be rearranged where the sequence of three of second level will remain unchanged. In the third level the sort will be applied on the values of the common three of second level. At the end of the process the three level non-decreasing data will become as the last format shown on the figure 3.

Now if the multilevel Binary Sort is applied on the data of the first table in figure 4 based on the ASCII value, the result will become as the same as the data of the second table in figure 4.

- **Before Using Multilevel Binary Sort:**

97;	ধ
97;165;	ধ্ম
97;168;	ধ্য
97;170;	ধ্র
97;170;402;	ধ্রূ
97;170;8220;	ধ্রু
97;376;	ধ্ব
98;	ন
99;	প

- **After Using Multilevel Binary Sort:**

98;	ন
97;	ধ
97;170;	ধ্র
97;170;8220;	ধ্রু
97;170;402;	ধ্রূ
97;168;	ধ্য
97;165;	ধ্ম
97;376;	ধ্ব
99;	প

Fig. 4. Multilevel Binary Sort is applied on the ASCII based Unicode data list.

C. Binary Tree and Binary Search Tree

Binary tree is a basic architecture in which each node has two own child. Binary Search tree is organized in a in-order binary tree. The tree may represent in linked data structure where each node is an object. Figure 5 shows a graphical representation of a complete binary tree based on the Binary Sorted (in order) data shown in figure 2.

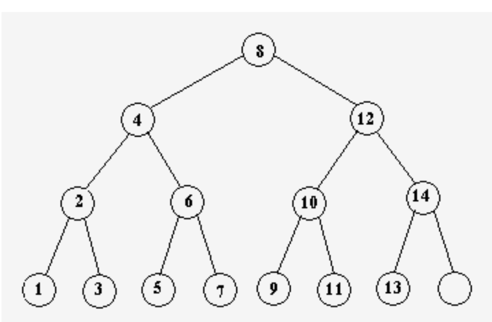


Fig. 5. A complete binary tree structure

Binary search tree has a property. If ‘x’ is a node in a binary search tree, ‘y’ is a left child of ‘x’, then y < x. If ‘z’ is right child of ‘x’, then z > x or z = x. The time complexity of binary search tree is O(h) where h is the height of the tree. In figure 5 the tree has height 4.

III. MULTILEVEL BINARY TREE

Binary search tree can be defined in two ways based on two conditions **z > x** and **z >= x** [here z and x are denoted as right and root nodes respectively]. But in the Multilevel Binary Tree structure only **z > x** is considered because all the same values of a single level of the Multilevel Binary Tree treated as a single node. If a desired value is found in any node of a level it will transfer its control to its branch node which holds the root of the next level tree of that node. The same condition is true for each node of a level of the Tree.

Suppose ‘T’ is a Multilevel Binary tree. Then T is called a Multilevel Binary tree if each node N of T has the following properties:

- N must have two leaf nodes and one branch node. Leaf nodes may hold sub tree of its level and the branch node must hold its next level tree of that node.
- The value of N must be grater than every value in its left sub tree (L) and must be less than every value in its right sub tree (R) i.e. **N > L** and **N < R.**
- If any value V is equal to N of that level, then V’s corresponding values will formed the next level tree with the same properties.

Figure 6 shows the 3 levels Binary tree formed as the values of the 3 level Binary Sorted data. First box indicates the first level tree then second level tree and finally third box shows the third level tree. On the tree 97 and 170 has no repetition to maintain the properties of the Multi Level Binary Search Tree (MLBST).

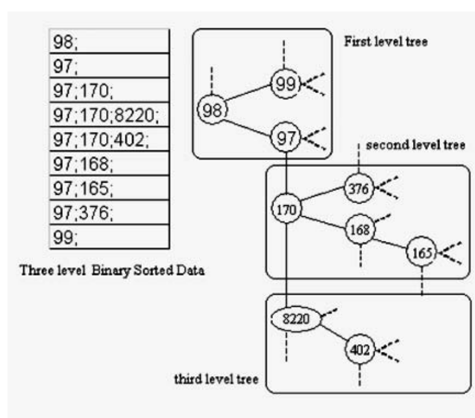


Fig. 6. Three level Binary Sorted Tree

A. Algorithm

Here first a MLBST_Node class is declared which holds the elements and links of the node.
The class, MLBST_Node, contains first two variables to hold searched Unicode data and last three values contains the MLBST_Node type object indicating the two leafs and a branch node.

The Buield_MLBST method builds the MLBST (Multilevel Binary Sorted Tree i.e. in order tree) from the Multilevel Binary Sorted data.

Declaring MLBST_Node **Class**
Start:
AsciiCode := 0 as a integer value.
UniCode := Null as a string value.
LeftNode := Null as a MLBST_Node type
RightNode := Null as a MLBST_Node type
BranchNode := Null as a MLBST_Node type
End:

Declaring Buield_MLBST **method with Five Parameters (**
Parameter1: Integer array containing the ASCII codes of a single character (simple/complex).
Patameter2: Integer value containing the length of the array
Parameter3: Integer value indicating the point of that array from where the ASCII codes will be used.
Parameter4: String value containing the corresponding Unicode.
Parameter5: MLBST_Node type value indicating the Parent Node.
)
Start:
If the 'Parameter3' is greater or equal to the 'Parameter2' then return TRUE.
If 'Parameter5' is equal to NULL then
Start:
AsciiCode := value of 'Paremeter1' at the position of 'Parameter3'.
If 'Parameter3' is equal to the 'Paremeter2' then
Start:
UniCode := value of 'Parameter4'
End:
Create a BranchNode of 'Parameter5'.
Increment 'Parameter3' by One.
Call Buield_MLBST method with the BranchNode of 'Parameter5' and return.
End:
If the value of 'Parameter1' at the position of 'Parameter3' is greater than
AsciiCode of 'Parameter5' then
Start:
If RightNode of 'Patameter5' is NULL then
Start:
Create a RightNode of 'Parameter5'
End:
Call Buield_MLBST method with the RightNode of 'Parameter5' and return.
End:
Else If the value of 'Parameter1' at the position of 'Parameter3' is less than
AsciiCode of 'Parameter5' then
Start:
If LeftNode of 'Patameter5' is NULL then
Start:
Create a LeftNode of 'Parameter5'
End:
Call Buield_MLBST method with the LeftNode of 'Parameter5' and return.
End:
Else If the value of 'Parameter1' at the position of 'Parameter3' is equal to
AsciiCode of 'Parameter5' then
Start:
If BranchNode of 'Patameter5' is NULL then
Start:
Create a BranchNode of 'Parameter5'
End:
Increment 'Parameter3' by One.
Call Buield_MLBST method with the BranchNode of 'Parameter5' and return.
End:
End:

Declare UniCodeTemp **as a string type global variable.**
Declare StartIndex **as an integer type global variable.**

Declaring Search_MLBST **method with Four Parameters** (
Parameter1: Integer array containing the ASCII codes of a single character (simple/complex).
Patameter2: Integer value containing the length of the array.
Parameter3: Integer value Indicating the point of that array from where the
ASCII codes will be used.
Parameter4: MLBST_Node type value indicating the Parent Node.
)
Start:
If the 'Parameter3' is greater or equal to the 'Parameter2' then return the **UniCode** of that Node.

If the value of ‘Parameter1’ at the position of ‘Parameter3’ is greater than AsciiCode of ‘Parameter5’ then
Start:
If RightNode of ‘Patameter5’ is NULL then return FALSE.
Else
Call Search_MLBST method with the RigntNode of ‘Parameter5’ and return.
End:

Else If the value of ‘Parameter1’ at the position of ‘Parameter3’ is less than
AsciiCode of ‘Parameter5’ then
Start:
If LeftNode of ‘Patameter5’ is NULL then return FALSE.

Else
Call Search_MLBST method with the LeftNode of ‘Parameter5’ and return.
End:

Else If the value of ‘Parameter1’ at the position of ‘Parameter3’ is equal to
AsciiCode of ‘Parameter5’ then
Start:
UniCodeTemp := UniCode of ‘Paremeter5’.
Increment StartIndex by One.
Increment ‘Parameter3’ by One.
Call Search_MLBST method with the BranchNode of ‘Parameter5’ and return.
End:
End:

IV. COMPLEXITY ANALYSIS

A. Build Tree

MLBST is like the Binary tree so each nod of this tree has two child nodes (Branch node holds its next level tree so branch node is not considered). Let the height of each level tree of the MLBST is ‘h’.
So, each level tree holds ($2^h - 1$) nodes.

Here each node has a branch node that holds the root of its next level tree. If the MLBST has two level trees then the total nodes become ($2^h * 2^h = 2^{2h}$). Here ‘-1’ is omitted for large value of ‘h’.

If MLBST has ‘n’ level tree then the total nodes will be 2^{nh} and complexity will be $O(2^{nh})$.

B. Search Tree

Let the height of each level tree is ‘h’ and the MLBST has ‘n’ trees. In worst case to search a desired data the complexity becomes $O(nh)$.

The best case occurs when the searched data exists in the root of the first level tree.

C. Complexity of MLBST applied on Bangla Unicode

In Bangla the total characters are approximately 300 including characters (vowel, consonant, complex character) and symbols. Again, for a char the ASCII code level is not larger than 4. Figure 7 shows a Bangla complex character having three level ASCII values and its corresponding UNICODE.

Complex Character	Three level ASCII	UNICODE
ম্প্ল	164;250;173;	ম্প্ল

Fig.7. Bangla complex character having three level ASCII.

Considering the ASCII level the MLBST has 4 level trees i.e. $n = 4$.

Let the first level tree holds the 300 nodes with the first level ASCII values. The each second level tree contains not more than 6 nodes of second level ASCII values (consider 10 rather than 6). Third and fourth level tree has a little number of nodes with the 3rd and 4th level ASCII values. Let the number is 4.

So, for the first level tree, $300 = 2^h - 1$. Taking log in both sides the equation becomes, $\log 301 = h \log 2$ i.e. $h = 8.233$. So for the 1st level tree it can be shown as $h_1 = 9$ (selling) i.e. height of the first level tree is 9. Similarly, for the 2nd level tree, $h_2 = 4$ (selling) and for 3rd and 4th level tree, $h_3 = h_4 = 3$ (selling). So the total nodes of the MLBST becomes,

$$(2^9 - 1) * (2^4 - 1) * (2^3 - 1) * (2^3 - 1) .$$

Omitting ‘-1’ the tree building complexity becomes $O(2^{19})$ and searching complexity becomes $O(19)$.

D. Representation of total number of nodes of a MLBST

Let the MLBST has ‘n’ number of level trees. If each level has a different size of height then the total number of nodes of the MLBST can be represented as

$$N =$$
$$(2^{h1}-1)*(2^{h2}-1)*(2^{h3}-1)*...*(2^{hn}-1)$$

Here N indicates the total number of nodes and 'h1' is the height of first level tree, 'h2' is the height of the each second level tree and so on. The equation can also be represented as,

$$N =$$
$$2^{\sum_{b=1}^{n} h_b} - \sum_{a=1}^{{}^{n}C_{n-1}} 2^{(\sum_{b=1}^{n} h_b) - h_a} +$$
$$\sum_{a=1}^{n-1} \sum_{m=a+1}^{n} 2^{(\sum_{b=1}^{n} h_b) - (h_a + h_m)} -$$
$$\sum_{a=1}^{n-2} \sum_{m=a+1}^{n-1} \sum_{n=m+1}^{n} 2^{(\sum_{b=1}^{n} h_b) - (h_a + h_m + h_n)} + \ldots$$
$$\ldots. \pm \sum_{a=1}^{{}^{n}C_{n-1}} 2^{h_a} \pm 1$$

In the last two terms have (+/-) together. It indicates that the even position of a term has the sign '-' (negative) and odd position of a term has the sign '+' (positive).

V. CONCLUSION

The algorithm is applied on Bangla Unicode but it may be applied on its related problem. It can also be used Unicode searching of other various languages. The main drawback of the algorithm is that, it needs a Multilevel Binary Sorted data to build the MLBST and its tree building complexity is $O(2^{nh})$ but MLBST build its tree only for first time.

ACKNOWLEDGEMENT

First of all, would like to express thank to almighty Allah who created us and give us the power to live in this world. Then thank to respected parents and my younger brother. Thank also goes to honorable teachers of IIUC.

REFERENCES

[1] Introduction to Algorithms (Second Edition) by. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein

[2] http://www.acm.uiuc.edu/conference/index.php

[3] http://www.jorendorff.com/articles/index.html

[4] http://www.unicode.org/unicode/reports/tr10/tr10-8.html

[5] http://www.connect-bangladesh.org/bangla/ webbangla.html

[6] http://www.betelco.com/bd/bangla/bangla.html

Centralized Management System for Mobile Communications with Pocket PCs

[1]M. Popa, [2]A.S. Popa, [3]V. Cretu, [4]M. Micea
[1,3,4]Computer and Software Engineering Department, Faculty of Automation and Computers
Department of Mechatronics, Faculty of Mechanical Engineering
University "Politehnica" Timisoara
2 Blv. V. Parvan, 300223 Timisoara, ROMANIA

***Abstract*-Pocket PCs are mobile computing devices largely used for mobile communications. Often Pocket PCs are long time used and modifications of the connection parameters to different networks are necessary. There is also need for installing different software packages. The mentioned operations can be manually done, by each user, this being a difficult and error prone solution or automatically, from a centralized location. This paper presents a centralized management system for mobile communications with Pocket PCs. It is useful for remotely setting connection parameters (dial-up phone numbers, SMSC numbers, DNS servers addresses etc.) and loading software packets to all or part of the Pocket PCs from a group leading to important saved times and the possibility to offer customized programs and graphical elements.**

I. INTRODUCTION

An accentuated spread of the mobile communications takes place in our days. More and more mobile devices are found in different domains: pagers, mobile phones, Pocket PCs etc.

Pocket PC, [1], is a mobile computer included in the category of Personal Digital Assistant (PDA). It accomplishes all the functions of an organizer (address list, calendar, meeting list etc.) plus other more complex applications such as: text editor, data base, mathematical operations, clock, data transfer with a PC etc. A Pocket PC has a RISC processor, an EPROM or Flash program memory, a RAM memory, a TFT or LCD display, with the usual dimension of 55x77 mm, Compact Flash, PCMCIA and Secure Digital card interfaces and several communication ports: serial, USB, IrDA, Bluetooth etc., each of them being useful for networking the Pocket PC.

They run an operating system such as Windows CE or Palm OS. There are many software applications running on PCs for Pocket PCs. Windows CE, [2], is an operating system used by a lot of embedded systems, including Pocket PCs. It offers to the application developers Win32 APIs, ActiveX controls, interfaces for message queues and other facilities as COM (Component Object Model), ATL (Active Template Library) and MFC (Microsoft Foundation Classes). The embedded system can be easily connected to PC through ActiveSync, using serial cable, network cable or infrared connection. The operating system offers support for multimedia (including DirectX), communication (TCP/IP, SNMP, TAPI) and security.

There are many situations in which Pocket PCs are long time used and modifications of the connection parameters to different networks are necessary. There is also need for installing different software packages. The mentioned operations can be manually done, by each user, this being a difficult and error prone solution or automatically, from a centralized location. This paper presents a centralized management system for mobile communications with Pocket PCs. It is useful for remotely setting connection parameters (dial-up phone numbers, SMSC numbers, DNS servers addresses etc.) and loading software packets to all or part of the Pocket PCs from a group.

The next section presents other similar works, the third section describes the created system and the last section outlines the conclusions.

II. STATE OF THE ART AND RELATED WORK

The mechanisms for centralized configuration of the systems and for remotely installing the applications are different for desktop PCs and for mobile devices.

For the centralized management of the PCs the Windows 2000 offers Group Policies. It helps the administrators to control the access of the users to different settings and applications through their affiliation to a group and not through individual settings of each user and compute.

For the mobile phones, OTA (Over The Air) Provisioning through WAP (Wireless Access Protocol) is a tool for configuring and remotely centralized installation of applications. Through OTA the administrators of the mobile telephony networks can control the parameters of the services and the software existent on the mobile phones. The Internet and the WAP standard are used for controlling the mobile phones. The first OTA configuration services were based on SMSs. Special SMSs were used, not shown to the user, interpreted by software existing on the mobile phone. The OTA configuration through WAP supposes the sending of a SMS Push through which a WAP page is specified and the phone will download it. The page may include setting elements and software packets which will be downloaded through WTP (Wireless Transfer Protocol). At the new mobile phone models the OTA configuration can be done through MMS (Multimedia Messaging Service) too. This service is offered to the users for sending images and sounds but can be used for sending also programs through SMIL (Synchronized Multimedia Integration Language). By using this language configuration parameters

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 327–332.

and software packets can be sent to a mobile device. The notification will be done with the same SMS Push mechanism as in the OTA Provisioning through WAP. In this case too, the configuration of a valid connection, including an Access Point Number, a Gateway Address, a MMS port number etc. is needed.

Reference [3] presents a Mobile Instant Messaging system, designed for mobile devices. It deals with the problem of configuring the mobile devices taking into account their mobility. Reference [4] describes the architecture of a universal manager that manages all mobile and non mobile devices from an enterprise. In reference [5] the problems of the implications and the requirements when TCP traffics are carried over UMTS that is characterized by moderate bandwidth bearers with relatively high link error, are approached. The implementation of the OTA service provisioning models in 3G environments is focused. The paper from reference [6] examines the characteristics of the mobile system and tries to clarify the constraints that are imposed on existing mobile services. It also investigates the enabling technologies and the improvements they brought. Reference [7] describes the implementation of Split Smart Messages (SSM), a middleware architecture similar to mobile agents that exploits dual connectivity on Smart Phones. Services can be executed, discovered and migrated on top of the SSM middleware.

Unlike the above mentioned achievements, this paper presents a centralized management system for mobile communications with Pocket PCs. It is useful for remotely setting connection parameters (dial-up phone numbers, SMSC numbers, DNS servers addresses etc.) and loading software packets to all or part of the Pocket PCs from a group.

III. THE CENTRALIZED MANAGEMENT SYSTEM (CMS)

The CMS has the following goals: the remotely configuration of the Pocket PCs, especially of their data communication parameters and the remotely installation of software packets, in CAB (Cabinet File) format.

The management of the devices is done centralized through a PHP program which allows the definition of profiles, the add of users in the defined profiles and the configuration of the connection parameters of the profiles and the add of the software packets which will be installed on the devices belonging to the profiles. The data introduced by an administrator in this program are memorized in a MySQL data base found on a web server. The data from the server are sent to the target devices by sending e-mail messages. In such a message a part of the URL address is sent in which either a HTML page is found, through which a user can send its options about the offered software packets or a XML document is found which describes the parameters to be set on the device and/or the URL addresses of the application packets which will be downloaded from the Internet and installed on the device.

On the Pocket PCs a number of applications will exist for receiving the messages sent by the administrator, for displaying the selection page of the optional programs, for processing the XML document and for downloading and installing the required software packets. For these operations the system uses the connection services to Internet offered by the Windows CE 3.0 operating system existing on the Pocket PCs.

A. The system's architecture

The system is made by several executable modules installed on the mobile device and by a web application manipulating a central data base. The modules of the application and their interaction are shown in Fig. 1.

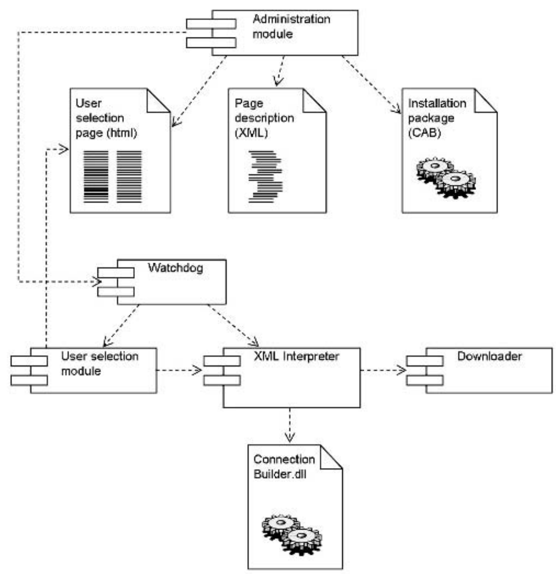


Fig. 1 The system's architecture

The Administration module is a PHP program through which an administrator can define groups (profiles) of users (Pocket PCs), can add users to the defined groups and can associate parameters to the profiles for different data connection settings (connections CSD/GPRS, WAP/Web/MMS, number of SMSC central etc.) and software packets required to be installed on the devices from a profile. The information introduced by the administrator and the associations between the profiles and the groups of parameters are memorized in a MySQL data base. When the administrator asks to the system to made the required modifications the Administration module announces the devices from the profile by sending them email messages, generates the HTML page for the selection of optional packets and, at the client's request, generates dynamically the XML document through which the required changes are described and the locations of the programs to be installed are specified.

All the modules, excepting the Administration module, are written in C++, using the SDK (Software Development Kit) specific to the Pocket PC platform. The modules are built with the help of the MFC framework and are compiled and linked using the tools from the eMbedded Visual C++ 3.0.

The Watchdog module is the component from the Pocket PC running continuously and waiting a message from the Administration module. When it arrives it extracts the useful information and launches either the User selection module for the optional packets (if the administrator's settings allow the user to choose between the optional applications) or the XML Interpreter (if the administrator has decided that the modifications will be done automatically without any intervention on the Pocket PCs).

The User selection module displays an HTML page generated by the Administration module for each profile. The application detects the users' options from the HTML page and generates the URL address where the PHP program generating the XML document is found. The parameters from the page and the information from the data base about the users, profiles and connection settings are used.

The XML Interpreter synthesizes the information described in the document generated by the Administration module and achieves the specified connections using the ConnectionBuilder.dll class library. The software packets mentioned in the document are downloaded on the device using the Downloader module and, next, installed.

The Downloader module downloads the files from the Internet and memorizes them in a local space. A progress bar and the displaying of downloaded percent show the progress of the copy operation.

The ConnectionBuilder.dll library contains classes modeling different data connection types and creates them using the services offered by the Windows operating system.

B. The Administration module

The Administration module is made by several modules. Each module contain its configuration page, the structure of the table from the data base, loading and saving functions in/from the data base and access functions to the internal data. Fig. 2 presents the modules and their connections.

The **index** module is the application's entry point being launched when the administrator accesses the configuration page. Its role is to load the next modules of the page. The **main** module is the main module of the page. The administrator accesses, creates and configures all the profiles starting from here. The **csd** module allows the administrator to define a profile with GSM CSD specific settings: access phone number, country code, network code, user name, password. These are memorized in the data base together with a name. The **gprs** module has the same functionality as the csd module but for GPRS connections. The **web** module is used for creating and configuring a connection to Internet. A web profile needs a csd or gprs profile. The **wap** module has a similar functionality the differences being at the configuration parameters for the WAP browsers installed on the Pocket PC. The **mms** module communicates with the MMS clients. It needs a data connection, usually GPRS, for communicating with the MMS gateway. Through the **email** module an administrator can create and configure a message store from the user's Outlook specifying its name, the POP3 and SMTP server. The **extra** module allows the software packets installation on the users' devices. The **target** module configures the users groups. It creates the groups considering more parameters: name, birthday, number of minutes spoken, the data traffic achieved in the last month. The users are accessed through their e-mail addresses. Each group can be associated with an existed profile. The **opr** module erases the profile associations from the data base and sends the e-mail messages for notifying the watchdog applications from the user's devices. The **connect** module contains the connection operations to the data base server, the user specification with its password, the opening of the data base associated to the application and the returning of the resulted object to the calling module. The **configure** module has the role of generating the XML document corresponding to the profile received as parameter. The profile's data are read from the data base. The generated document is read by a web browser or by the XML Interpreter.

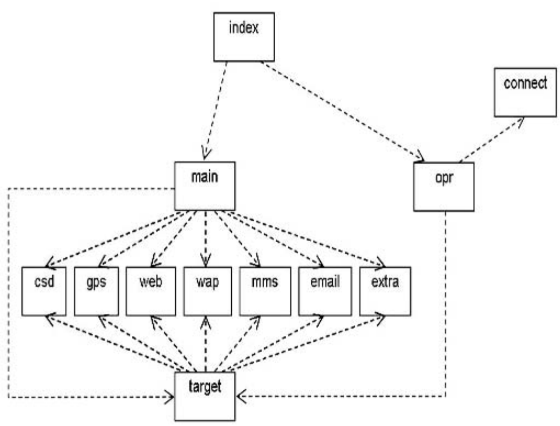


Fig. 2 The Administration module

The administrator accesses the page generated by the index module, This page loads also the main module and through it the other modules too which read the existing profiles from the data base. Next, the administrator creates new profiles or modifies the parameters of the existing ones and saves them in the data base. At this moment the selection page through which the user will specify the desired software packets will be generated. The last step is to send notification e-mails to all the users from the group already defined. This message will contain the identification recognized by the watchdog application and the part of the URL identificator which will help to create the way where the selection page is found or the XML document from which the information for configuring the device are read.

C. The Watchdog module

The Watchdog module waits a message from the Administration module and if it receives one with an expected format it will announce an other module according to the type of the message: the User selection module if it is desired that the user selects one or several from the optional software packets or the XML Interpreter if the intervention of the user is not required. The capitation of a message, its interpretation and the initiation of the action required by it are done by different classes. The architecture of the module, showing the classes, the interfaces and the communication channels between the objects is shown in Fig. 3.

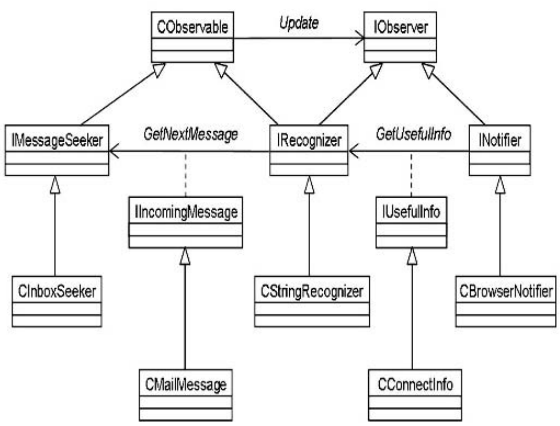


Fig. 3 The architecture of the Watchdog module

D. The User selection module

Its goal is the user intervention on the installing process. This module is launched in the case of an installation in which the user has the possibility to choose zero, one or more from several optional software packets (graphical interfaces, ring tones etc.) for being downloaded from Internet and installed.

For offering maximum flexibility concerning the graphical user interface an interface based on the HtmlView control available through the API from Win32 was thought. It has an aspect similar to that of an Internet Explorer window (Pocket Internet Explorer in this case). This window can be integrated in the window created by the programmer. The HtmlView window accesses a page found at the address received in the command line and displays it to the user. After the user marks the desired packets he will set the button for accepting the selection or for canceling it. If the selection is accepted the address of the PHP page together with the parameters are sent in a command line to the XML Interpreter and if not the installing process is stopped. Fig. 4 shows the class diagram of the User selection module.

E. The XML Interpreter module

The scope of this module is to read a XML document, to interpret the contained information and to achieve the required actions mentioned in the document. The location where the document can be found is received as a parameter in a command line from appealing module. It can be the Watchdog module in the case of an automate configuration/ installation or the User selection module if the user has to decide which software packets will be installed on its device. Fig. 5 shows the class diagram of this module.

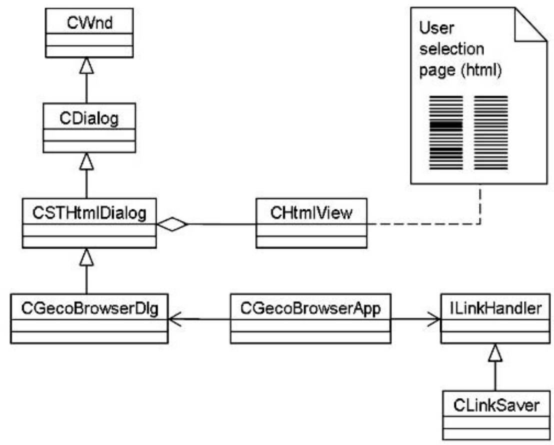


Fig. 4 The architecture of the User selection module

The IBuilder type object will generate a number of ISetting type objects. It is the interface for the objects which contain the functionality of creating settings or installing programs.

The CCsd class encapsulates the attributes of a data connection through CSDIt contains data concerning the parameters of the serial port at which the modem is connected.

F. The Downloader module

The scope of this module is to download files from Internet, or any other network supporting HTTP connections) indicating in a graphical way the progress of the operation. The file URL address and its place in the local files system are sent to the module as parameters in the command line.

The application has two threads: for reading the information from the specified URL location and writing in the file from the local files system and for displaying the window showing the progress of the operation. The classes from the two threads communicate through the Observer mechanism. The thread which copies the file uses CFile, CInternetSession or CHttpFile objects and the other thread has classes which use and are derived from GUI classes: CDialog, CProgressCtrl. The parts of the program accessing variables belonging to both threads are protected with CSingleLock type semaphore objects. Fig. 6 shows the class diagram of the Downloader module.

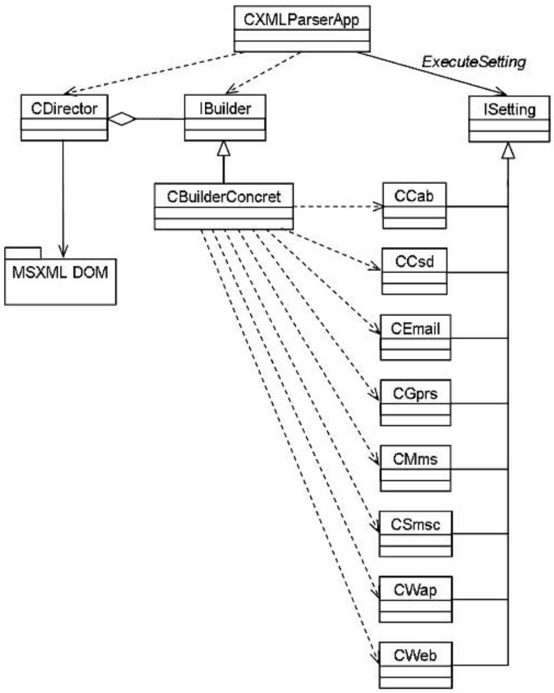


Fig. 5 The architecture of the XML Interpreter module

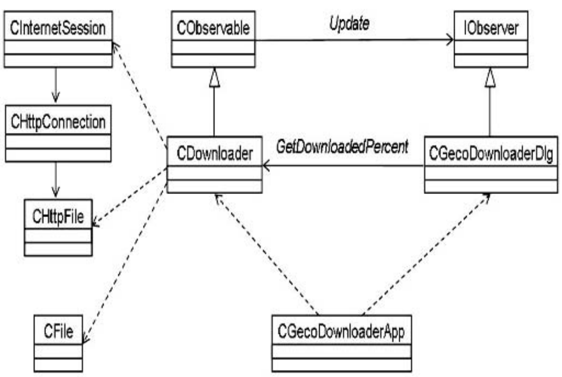


Fig. 6 The architecture of the Downloader module

The CGecoDownloaderApp class is the entry point of the application, in accordance with the functioning model of the MFC framework. From this class the dialog object is created, the downloader object is created, by firstly calling the GetDownloader method, the file copying is started and the displaying of the downloading operation progress is started.

The classes CInternetSession, CHttpConnection and CHttpFile are parts of the MFC library which creates a connection between a client and a HTTP server allowing the access of the file from the server as from the local files system. Their use permits the reading from the file from the server, the writing in it (if the needed permissions exist) and the finding of its attributes (dimension, creation date etc.)

The CDownloader class encapsulates the functioning of creating a connection to the HTTP server, the sending of the requests to it, the determination of the dimensions of the desired file, the reading of the information from the remote file and its saving to the local file. The CDownloader class extends the CObservable class, ensuring the notification mechanism of the classes involved in the progress of the download (the NotifyAll method is called after copying a fixed part of the file dimension).

The download thread runs the DownloadThread statical method of the CDownloader class. This thread is launched at the end of the DownloadFile function, called by the application object, after creating the Internet session, sending the request to the HTTP connection and opening of both the files.

The CGecoDownloaderDlg class is responsible by the graphical interface of the application. It implements the IObserver interface for being notified by the CDownloader object. For displaying the information the class uses a CProgressCtrl object and a static control text in which the percent of the file downloading progress is written. When the downloader calls the Update method the object reads the new value of the percent through the function GetDownloadedPercent function and modifies the dialog controls.

The IObserver and CObserver classes offer the Observer mechanism used for structuring the whole module. The CObservable class implements the retention and notification of the observers and the IObserver interface offers the Update function for being called in the subject class.

G. The ConnectionBuilder module

The ConnectionBuilder module ensures dial-up data connections and GPRS connections for external modems (in the case of mobile phones) or internal modems (in the case of Pocket PC Phone Edition device). For creating the required connections the classes of the library use the functions of the Remote Access Service API. Fig. 7 presents the class diagram of the ConnectionBuilder module.

Only the IConnection class is exported from the library, its clients having not access to the other classes and structures. It defines a generic interface for configuring parameters as the server phone number, the IP addresses of the DNS addresses (if it is the case), user name, password etc. Also a method returning to the client an IConnection interface type object through which the connection will be created after the configuration of the parameters exists in this class.

The CGenericConnection class contains the common parts of all the setting types found in the module. These parts are linked to the creation and configuration of the RASENTRY and RASDIALPARAMS structures through the functions offered by the IConnection interface. These structures exist in any RAS connection. When the class instance is created the members of the structure will be configured for needing the smallest number of parameters (meaning the chose of the modem type and of the phone number). The other parameters receive implicit values for a minimum intervention from the user part. The CCsdConnection class is responsible for configuring and creating a CSD (Circuit Switched Data) dial-up connection. Its particularities are given by the DEVMINICFG structure which has as purpose the configuration of the connection to the modem (the serial transfer speed, the character's number of bits etc.).

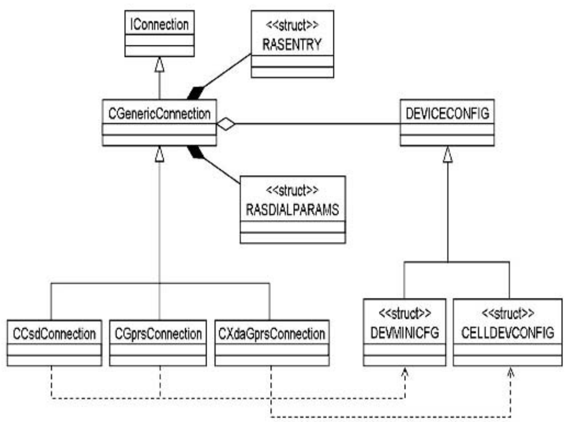


Fig. 7 The architecture of the ConnectionBuilder library

The CGprsConnection class localizes the particularities of a General Packet Radio Service connection (for the modems and the communication networks allowing this service) achieved by an external modem. For this service an APN (Access Point Name) must be defined which contains the URL address of the gateway at which the device is connecting. An other particularity of the GPRS connections is the including of the APN's name in the extra dialog string (the character string sent to the modem before the connection request at APN) in the form `+cgdcont=4,"IP","apn_name_,'',0,0.` The class uses the same structure DEVMINICFG as in the CSD connections cases, this structure being specific to the communication with an external modem.

The CXdaGprsConnection class has the role of creating a GPRS connection for the internal modem, situation in which the configuration of the device is made with the CELLDEVCONFIG structure from the TSP (telephony Service Provider) API.

V. CONCLUSIONS

There are two major application areas for the described system: the mobile telephony networks and the companies using Pocket PCs in their activities.

If the system is used for managing the subscribers of a mobile telephony network, the most important advantages are:

- automatic configuration of the connecting parameters for achieving a data connection; a company which offers data communications services for many clients, owners of Pocket PCs, will be able to migrate to the new settings without quality problems and without involvement of the clients;
- the possibility of offering customized programs and graphical elements.

If the system is used by a company working with many Pocket PCs the advantage consists in the time saved when the needed applications are paralelly installed on all or the desired mobile devices.

As disadvantage, the necessity of installing the client part on each Pocket PC can be mentioned. This installation will need also the configuration of a first connection to Internet for the mechanisms of notification and downloading of the selected software packets.

The system can be extended for creating a monitoring way for the state of the Pocket PCs. The administrator could verify how many users, from a specified group, have accepted the offered software tools, what setting problems occurred, how many users have problems with the client software etc. Facilities for generating statistical rapports concerning the utility of different programs and graphical interfaces could be also added.

Another development direction could be the achievement of the connections in several steps for avoiding the limitation given by the need of a connection to Internet. The notification could be achieved through SMS, for specifying the parameters needed for a data connection with a server for the existing mechanism, followed by the OTA configuration.

REFERENCES

[1] Pocket PC Developer's Network, http://www.pocketpcdn.com
[2] N. Gratten, M. Brain, *Windows CE 3.0 Application Programming,* Prentice Hall, 2000, ISBN-10: 0-13-025592-0, ISBN-13: 978-0-13-025592-1
[3] R. Parviainen, P. Parnes, „Mobile Instant Messaging", in *Proc. of The 10th International Conference on Telecommunications, ICT'2003,* Tahiti, February – March 2003
[4] S. Adwankar, S. Mohan, V. Vasudevan, „Universal manager: Seamless Management of Entreprise Mobile and Non-Mobile Devices", in *Proc. of 2004 IEEE International Conference on Mobile Data Management,,* Berkeley, California, USA, January 2004
[5] N. Zhang, „Simulation-based investigation of TCP file transfer over UMTS", University College London 2003
[6] I. Jorstad, S. Dustdar, D. Van Thanh, „Evolution of Mobile Services: An analysis of current architectures with prospect to future", in *Proc. of The 2nd International Workshop on Ubiquitous Mobile Information and collaboration Systems (UMICS),* Riga, Latvia, June 2004, Springer LNCS, ISBN: 3-540-24100-0
[7] N. Ravi, C. Borcea, L. Iftode, „Split Smart Messages: Middleware for Pervasive Computing on Smart Phones", Rutgers University Technical Report DCS-TR-565, 2005

A Navigation Tool for Blind People

Mounir Bousbia-Salah, Mohamed Fezari
University of Annaba
Faculty of Engineering
Department of electronics
Laboratory of Automatic and Signals-Annaba
BP.12, Annaba, 23000, ALGERIA
Phone/Fax:+213 38 876565
E-mail: mbousbia_salah@yahoo.com, mouradfezari@yahoo.fr

***Abstract*-This paper describes the development of a navigation aid in order to assist blind and visually impaired people to navigate easily, safely and to detect any obstacles. The system is based on a microcontroller with synthetic speech output. In addition, it consists of two vibrators, two ultrasonic sensors mounted on the user's shoulders and another one integrated into the cane. This aid is able to give information to the blind about urban walking routes and to provide real-time information on the distance of over-hanging obstacles within six meters along the travel path ahead of the user. The suggested system consists then in sensing the surrounding environment via sonar sensors and sending vibro-tactile feedback to the user of the position of the closest obstacles in range. For the ultrasonic cane, it is used to detect any obstacle on the ground.**

I. INTRODUCTION

Mobility is one of the main problems encountered by the blind in their daily life. Over time, blind and visually impaired people have used some methods and devices, such as the long white cane and guide dog, to aid in mobility and to increase safe and independent travel. Due to the development of modern technologies, many different types of devices[1] are now available to assist the blind to navigate. There are commonly known as electronic travel aids. Among these aids are Sonic Pathfinder [2], Mowat- Sensor [3], and Guide-Cane [4] which are called clear path indicators or obstacle detectors since the blind can only know whether there is an obstacle in the path ahead [5]. These devices are used to search for obstacles in front of the blind person, and they operate in a manner similar to a flashlight, which has very narrow directivity. Sonic-Guide[6] and NavBelt [7], however, are called an environment sensor since it has wide directivity enabling it to search for several obstacles at the same time.

The purpose of this project was to create a prototype of a device that can help blind people to travel with increased independence, safety, and confidence.

The proposed system involves a microcontroller with speech output. It is a self contained portable electronic unit. It can supply the blind person with assistance about walking routes by using spoken words to point out what decisions to make.

In addition, and in order to overcome the imperfections of existing electronic travel aids, the suggested method of measuring distance travelled in this system, is to use the acceleration of a moving body which in this case is the blind person. An accelerometer, followed by two integrators is used to measure a distance travelled by the blind. This technique is considered in inertial navigation systems [8] and suffers from drift problems caused by the double integration and offset of the accelerometer which are overcome by the footswitch [9]. When this footswitch is closed, the acceleration and the velocity are known to be equal to zero and this can be used to apply a correction.

In order to help blind travellers to navigate safely and quickly among obstacles and other hazards faced by blind pedestrians, an obstacle detection system using ultrasonic sensors and vibrators has been considered in this aid. The proposed system detects then the nearest obstacle via streoscopic sonar system and sends back vibro-tactile feedback to inform the blind about its localization. On the other hand, an ultrasonic cane equipped with wheels is considered to detect any obstacle which may be on the ground.

The system has then an environment recognition and a clear path indicator functions.

II. REQUIREMENTS

Portability, low cost, and above all simplicity of controls are most important factors which govern the practicality and user acceptance of such devices.

The electronic travel aid (ETA) is a kind of portable device. Hence it should be a small-sized and lightweight device to be proper for portability.

The blind is not able to see the display panel, control buttons, or labels. Hence the device should be easy to control: No complex control buttons, switches and display panel should be present. Moreover, the ETA device should be low-price to be used by more blind persons.

Our system is developed for portable (small size and lightweight), inexpensive and easy to use, and low-power consumption (supplied by battery).

III. PRINCIPLE OF OPERATION

The aid consists of a microcontroller as processor, an accelerometer, a footswitch, a speech synthesizer, an hexadecimal keypad, a mode switch, three ultrasonic sensors, two vibrators and a power switch. Fig. 1. shows the block diagram of the system.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 333–337.

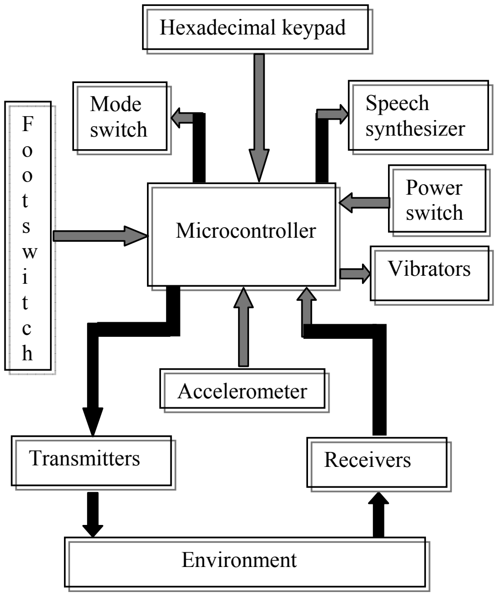


Fig. 1. Block diagram of the system.

The obstacle detection part of the system contains three ultrasonic transmitters-receivers and two vibrators.
Two pairs of these ultrasonic sensors are mounted on the blind's shoulders[10] as shown in fig. 2.

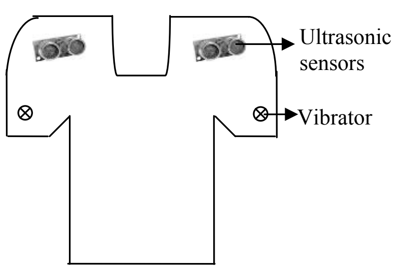


Fig 2. Sonars mounted on shoulders.

The other is cane type subsystem[11] as shown in fig 3. It is equipped with ultrasonic sensors and wheels. The user walks with holding this cane type system in front of him like the white cane. The cane type system notifies whether any obstacle is in the middle of the walking direction. Since the wheels are always contacted with ground, the user can recognize the condition of ground such as depression, cavity, and the stairs with his hand's tactile sensation intuitively.

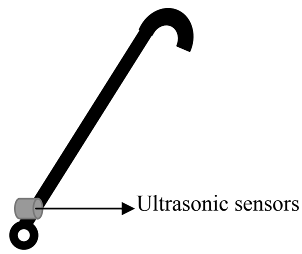


Fig 3. The ultrasonic cane.

This obstacle detection system use a 40 KHz ultrasonic signal to acquire information and can detect the presence of any obstacle within the specified measurement range of approximately 0.03 to 6 meters. It operates by sending out a pulse of ultrasound. Eventually the pulse is reflected from a solid object in the path of the pulse. The time between the outgoing pulse being transmitted and its echo being received corresponds to the distance between the transmitter and the object or the obstacle. This information is then relayed to the blind in some vibro-tactile way and speech way(for the cane).

On the other hand and as the 'Micromap'[12], the system has two modes of operation, record and playback. In addition, the playback mode has two directions, forward and reverse. The user selects then, one of these three possibilities by a switch.

In the record mode, the blind walks the route of interest, and the aid measures the distance travelled by the user. When the blind reaches a decision point, for instance a point at which the route takes a left turn, the user presses a key on the aid coded with a left turn instruction. This has two effects:
- The distance travelled is stored in memory of the ;microcontroller, and the counter reset to zero.
- The left turn instruction is stored.

Afterwards, the blind walks to the next decision point and the above procedure is repeated.

In the playback mode, the aid measures again the distance travelled by the user. When this is equal to that stored in the memory for that particular section of the route, an audible signal is given to the blind. The audible signal is coded to indicate what action the user should take at this point, for instance turn left. In the reverse direction, the procedure is exactly the same except that the route information stored in the memory is used in reverse order, and that right and left are interchanged.

At decisions points, the blind can make any of the following decisions:
Turn right; Turn left; Cross road; Cross road junction; Pedestrian crossing; Steps; Pause; Stop.
Each of these decisions has separate key. There are also two extra keys available, which are undefined in the present software, but which the blind could have available for their specific use.

The system can store a number of routes, each of which is numbered, and be selected using the same set of keys as for the decisions. In practice the number is likely to be set by the size of the available memory.

IV. Technical Description

This section describes in some detail the components of the proposed navigation aid.

A. Microcontroller

The microcontroller used in the aid is the PIC 16F876[13] from 'MICROCHIP', with 8 k of 14 bits program memory, 368 bytes of RAM and 256 bytes of data EEPROM.

B. Accelerometer

The accelerometer used is the ADXL213[14] from 'Analog devices'. It has a range of ±1.2 g and a sensitivity of 30%/g. With this accelerometer, no A/D converter is then required as the output is digital.

On the other hand, the accelerometer needs to be attached to the shoe or to a rigid part of the leg where the condition of both acceleration and velocity equal zero is applied.

C. Speech synthesizer

The speech synthesizer device chosen is the ISD 5216[15] from 'ChipCorder' and is used as an audio output. The chip is a single-chip solution offering digital storage capability and up to 16 minutes of high quality, audio record and playback functionality.

On the other hand, the speech synthesizer is activated by pulses from the microcontroller. The output represents the different actions to be taken (e.g. road right turn, left turn…). The speech synthesizer chip with a small vocabulary tells then the blind person about travelled distance, present location and decisions to make. Information about the route is stored in the memory in the form of a digital map of the device to guide the user to his destination via the planned routes.

For obstacle detection, an increase of distance to an obstacle results in a decrease in vibration, while a decrease of distance results in an increase in vibration.

D Headphones

Since hearing for blind people is very important, the headphones would dull this sense. For this system, it has been decided to consider headphones used for walkman. Spoken words from the speech synthesizer which represent the different action to be taken will therefore be heard by the blind.

E. Hexadecimal keypad

In order to input information an hexadecimal 4x4 keypad is used in this aid. It is placed on the side of the case, and can be seen in fig. 4. The keypad switches enable the user to select routes and to enter decision. It is of course possible to label these keys with Braille symbols if it is thought necessary.

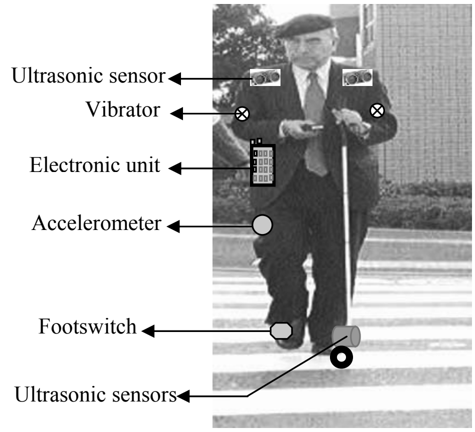


Fig. 4. The navigation aid worn by the blind.

F. Footswitch

The footswitch is used to allow the PIC 16F876 to provide frequent corrections of drift effects. This footswitch 'S' needs to be attached to the heel of the shoe. When the blind starts to walk, 'S' is equal to zero. The microcontroller estimates then the acceleration and calculates the distance.

When the footswitch is on the ground, 'S' is equal to one. The microcontroller estimates and calculates the errors. Afterwards, corrections are made.

The micro-switch is one example of switch which can be used because it is more flexible.

V. Obstacle Detection

For obstacle detection and as aformentioned, the aid is provided with an ultrasonic system attached to the jacket and an ultrasonic cane. It is is based on three ultrasonic sensors and two vibrators.

A. Ultrasonic sensors

The sonar system is based on two ultrasonic sensors mounted together. One emits an ultrasonic wave while the other measures the echo. By differentiation of the input and output signals, the PIC16F876 computes the distance to the nearest obstacle. Then this information is transmitted as a Pulse Wide Modulation (PWM) signal to the receiver.

The ultrasonic module used as sensor for this application is the MSU10[16] from 'Lextronic' and can be seen in fig. 5.

It has the following characteristics:

- Angle of detection: approximately 72°.
- Dimensions: 32 x 15 x 10 mm.

Fig. 5. The ultrasonic sensor.

B. Vibrators

In this system, vibrators from mobile phone technology have been used. Those devices are small and light enough to be fixed on cloth without any obstruction.

C. Ultrasonic Cane

The ultrasonic cane used for this system is based on an ultrasonic transmitter-receiver which detects obstacle on the ground.

D. Data treatment

The microcontroller gathers the information from the ultrasonic sensors as PWM signal directly proportional to the distance of the nearest obstacle. Afterwards, it measures the width of the transmitted pulses and converts it into empiric distance.

Following a calibration phase, the real distance between the sensor and the obstacle can determined . The direction is given by comparison of the signal from both sensors. This distance is then converted into a voltage command for appropriate vibrating feedback. The system redirects this information to the actuators via Serial Peripheral Interface. A multichannels D/A Converter recovers 2 integers (address and data) and sends the desired output voltage to the appropriate vibrator.

VI. Use Of The System

The system is easy and straightforward to use. It is attached to a belt which is fastened around the user's waist. There is provision for a test to ascertain that the blind person's step is detected by the accelerometer. The user then selects the route number, and the appropriate mode and direction.

A repeat key has been considered to enable the blind person to make the aid repeat the word indicating a decision. This is to ensure that the user can be certain of the decision, in case it is obscured the first time by, for example, traffic noise.

On the other hand, when an obstacle is detected, vibrotactile output occurs in pulses at a rate inversely related to the distance from the user. If there is no obstacle detected, no vibrational pulses are emitted.

In addition, the blind should know from which direction the obstacles are coming from. Localization on the horizontal plane is done by appropriate combination of vibration between the left and the right side. If the user feels a vibration on its right it means that the obstacle is on his right and vice versa. If the vibration is on both sides the obstacle is in front of him.

VII. Experiment Results

The first field trial of the route planning was tested on two blind persons. The test routes were of about 100 metres along roads and the results are shown in fig. 6.

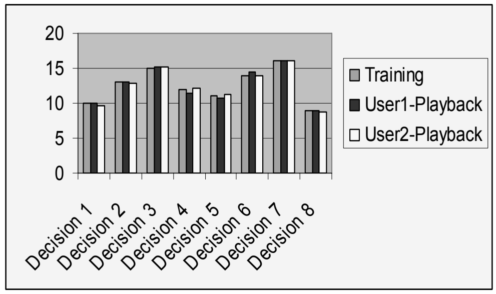


Fig. 6. Results of the test.

It can be seen a minor discrepancy from these results for the following two possible reasons:

- The aid may not have been correctly adjusted to detect every step.
- The user may have had a significantly different gait between the record and playback modes.

VIII. Conclusion

This paper has presented the development of a navigation aid in order to enhance the independent mobility of blind individuals. The technique well known in aircraft navigation used in this study has reduced errors caused by the accelerometer and double integration. The use of the footswitch is also highly advantageous because without it, drift errors due to the accelerometer and double integration would be considerably greater in magnitude and would reduce the effective range of the electronic travel aid.

The system has been used on some preliminary trials. The results obtained are encouraging and in the near future, it is planned to carry out more extensive tests.

Although the system detects the nearest obstacle, it cannot solve the blinds' ultimate problem of the environment perception. It has limits due to the characteristics of the ultrasound reflections such that many object can barely be detected, which have very small or soft surfaces. Despite these difficulties, it is hoped that the proposed system will efficiently aid the blind at navigation.

For future development, and as it is difficult to know where the blind is globally, it is then desirable to use the global positioning system (GPS)[17] in order to get the user position information.

Acknowledgment

The authors would like to thank the laboratory of automatic and signal at Annaba for its contribution in the development of this project.

This page intentionally blank

REFERENCES

[1] E. Pissaloux, *Compensation de la déficience visuelle. Systèmes d'aides à l'handicap* (Pruski, Ed.), Hermes, 2003.
[2] A. Dodds, D. Clark-Carter, and C. Howarth, "The sonic PathFinder: an evaluation," *Journal of Visual Impairment and Blindness,* vol. 78, no. 5, pp. 206–207, 1984.
[3] A. Heyes, "A polaroid ultrasonic travel aid for the blind," *Journal of Visual Impairment and Blindness*, vol. 76, pp. 199–201, 1982.
[4] I. Ulrich, and J. Borenstein, "The guide cane-Applying mobile robot technologies to assist the visually impaired," *IEEE Transaction on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 31, no. 2, pp. 131-136, 2001.
[5] T. Ifukube, T. Sasaki, and C. Peng, "A blind mobility aid modeled after echolocation of bats," *IEEE Transactions on Biomedical Engineering*, vol. 38, no. 5, pp. 461–465, 1991.
[6] J. Barth, and E. Foulhe, "Preview: A neglected variable in orientation and mobility," *Journal of Visual Impairment and Blindness*, vol. 73, no. 2, pp. 41–48, 1979.
[7] S. Shoval, J. Borenstein, and Y. Koren, "The NavBelt- A computerized travel aid for the blind based on mobile robotics technology," *IEEE Transactions on Biomedical Engineering*, vol. 45, no 11, pp. 1376-1386, 1998.
[8] J.F. Fournier, and J. Hamelin, "Problèmes de navigation aérienne au pôle nord," *8ème colloque international du centre d'études arctiques (CNRS)*, Paris, pp. 179-187, 1983.
[9] M. Bousbia-Salah, A. Larbi, and M. Bedda, "An approach for the measurement of distance travelled by blind and visually impaired people," in Proc. *10th IEEE International Conference on Electronics, Circuits and Systems*, Sharjah, United Arab Emirates, pp. 1312-1315, 2003.
[10] S. Cardin, D. Thalmann, F. Vexo, "Wearable obstacle detection system for visually impaired people," *VR workshop on haptic and tactile perception of deformable objects,* Hannover, Germany, 2005.
[11] Y.J. Kim, C.H. Kim, and B.K. Kim, "Design of auditory guidance system for the blind with signal transformation from stereo ultrasonic to binaural sound," in *Proc. Int. Sym. Robotics,* 2001.
[12] I.L. Freeston, V.L. Callaghan and N.D. Russel, "A portable navigation aid for the blind," *IEEE frontiers of engineering and computing in health care*, pp. 247-249, 1984.
[13] C. Tavernier, *Applications industrielles de PIC*, ETSF, Paris, 2001.
[14] Analog Devices, *ADXL213 Datasheet*, Norwood, MA, 2004.
[15] ChiCorder, *ISD5216 Datasheet*, 2003.
[16] Lextronic, *MSU08 Datasheet,* 2005.
[17] H. Hashimoto, K. Magatani, K. Yanashima, "The development of the navigation system for visually impaired persons," *in Proc. 23rd Annual EMBS international conference*, Istanbul, Turkey, pp.1481-1483, 2001.

Mobile Computing Tools as a Medium to Educate and Empower people with Chronic Conditions

Abhilash Geo Mathews*, Dr Richard Butler*, Dr Joan Love
* Robert Gordon University, Aberdeen
0100025@rgu.ac.uk, r.a.butler@rgu.ac.uk, j.love@rgu.ac.uk

Abstract : A wearable health care monitoring system is proposed. Constant health monitoring can improve the patient's quality of life for various health conditions such as diabetes and obesity, the focus being on prevention rather than treatment. To achieve this individuals and patients need to be empowered and educated with the use of proactive mobile computing tools and technologies such as mobile phones, PDAs, Bluetooth and WAP. Integrating these tools provides a transparent way of monitoring, analysing and modelling their metabolic performance and allows patients to become more responsible for the management of their health conditions.

I. OBJECTIVE

This project aims to empower and educate individuals, patients having chronic conditions(e.g. obesity, diabetes) by providing them with proactive mobile computing tools. These tools will allow them to maintain control of their condition by monitoring, analysing and modelling their metabolic performance and how their condition affects it. The motivation behind this work lies in the pandemic of chronic conditions in the developed world and similar increases in the developing world.

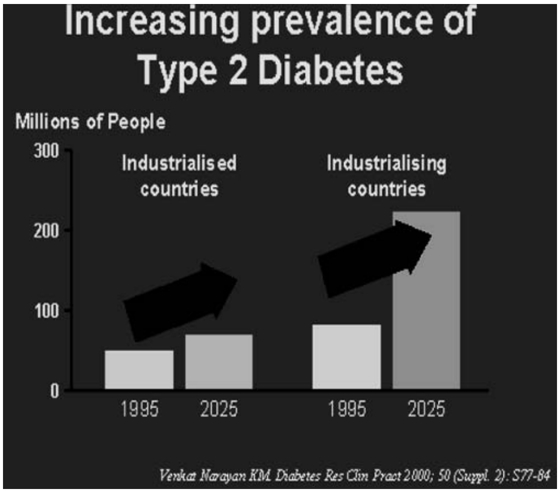


Fig 1: Increasing prevalence of Type 2 diabetes

Today, there are over 171 million people with diabetes [1] and over a billion people overweight, of which over 300 million are clinically obese [2]. With future estimates revealing an increasing prevalence of chronic conditions in both the developed and developing countries (Fig 1), healthcare providers are looking to alternative mechanisms for delivery of care, and by which the quality of care, quality of life and life expectancy can be improved for the patients. With long waiting lists to receive health services, patients need to become more responsible for the management of their conditions and require tools to empower and to educate them to achieve this.

II. BACKGROUND

This paper uses, as an example, the issues surrounding the care of people with diabetes.

There is, without doubt, a severe personal cost to the individual with diabetes. In addition to this heath care providers, such as the NHS, are under severe pressure from the growing burden of treating people with diabetes on a daily basis and dealing with the resulting complications. Also, the shortage in medical professionals such as doctors and nurses has further escalated the problem of faster health care delivery.

In 1997 treating diabetes and its complications cost an estimated 9% of the healthcare budget in the UK. By 2011 that figure is projected to rise to nearly 25% [3]. The NHS across the UK will spend £65.4 billion in 2002-03 [4]. Thus the estimated spend on diabetes and resulting complications is between £5.9 and 10.4 billion. The trend of escalating diabetes prevalence will lead to an immense financial burden on the health service unless preventative measures are taken to control the incidence of diabetes and its complications.

The Diabetes Control and Complications Study (DCCT) carried out with people with type 1 diabetes in the US and Canada proved to be a landmark in diabetes care [5]. The 1998 United Kingdom Prospective Diabetes Study profoundly changed the UK perspective on the management

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 339–343.

of diabetes [6]. Both studies proved that the level of blood glucose control predicts the onset and severity of diabetes-related complications in diabetes. This means that if a person with diabetes can keep blood glucose levels as close as possible to normal, they can live a normal life span with few, or even no, complications.

In the past good control was achieved through strict regimen and prohibitions. The present, and future, emphasis is on empowering the person with diabetes to manage their condition themselves. In recent years, new formulations of insulin have become available which have been designed to offer the advantages of simpler regimens and better glucose control. These are varied according to the diet and activities of the person with diabetes in an attempt to permit more normal patterns individualised to a persons own habits.

In practice the onus of maintaining good control can be overwhelming. At best, it is a constant balancing act of monitoring blood glucose, keeping control of diet and exercise and calculating how much insulin is required. At worst, people with diabetes can become so disheartened that they feel unable to cope.

The growing number of people in the UK with diabetes, the evidence from the DCCT [5] and UK PDS [6] that tight control of blood glucose levels improves life expectancy and reduces complications in diabetes and evidence that supported self-care improves outcomes has prompted the publication of a National Service Framework (NSF) for Diabetes for England and Wales [7] in conjunction with the Scottish Diabetes Framework [8]. These documents outline the infrastructure to support the drive to raise standards of diabetes care. A key target is for hospital-based clinics to implement an effective IT system, which will allow them to support all aspects of diabetes care.

NHS Scotland [9] provides the following definitions relating to such a system:

"Information Management & Technology (IM&T) is about the information which NHS Scotland needs to deliver effective healthcare, the technology needed to deliver that information to the right person at the right time, and the range of processes such as training and support services needed to make it happen."

"eHealth encompasses much more than the deployment of computer technology. It conveys the message of electronics in support of health and stimulates thought and discussion about the broad range of issues and opportunities that technology offers in the health care setting to both healthcare professionals and patients."

Mobile computing tools, as described in this paper, would be a key component in such IM&T and eHealth systems.

The challenge lies in developing innovative, validated, decision support tools to aid diabetes self-management [10]. Such decision support tools may help patients and their families optimise blood glucose control and reduce the long-term complications associated with poor control of diabetes.

A. Computer based, personalised, diabetes simulator

The mobile computing tools, as described in this paper utilise, and extend, previous work on a computer based, personalised, diabetes simulator in diary format developed as a predictive tool for patients. This simulator was devised with the aim of helping reduce the 'trial and error' involved in diabetes management by allowing patients to simulate and experiment with dietary or insulin adjustments using a software 'body-double' on a desktop or laptop PC.

This 'body double' concept, and underlying mathematical model, was evaluated and refined during 2000/2001 in a field trial involving 43 people with Type 1 Diabetes. The trial successfully proved that under normal conditions the models involved adapted to the individual's metabolism and predicted blood glucose level. [11]

In 2003/4 a further trial was undertaken, involving 19 children with diabetes. The children and their families used the body double software for a number of days. This allowed them to become familiar with the software and for the body double personalisation process to occur. The children then continued to use the software while wearing a MiniMed™ Continuous Blood Glucose Monitor (CGMS). The blood glucose profiles predicted by body double software and measured by the CGMS were then compared. The modelled values of blood glucose were found to correlate well with the CGMS data. The concurrent CGMS recordings provide a large data set to modify and improve the model. Patient feedback has also led to improvements in the usability of the software [12].

The underlying mathematical model is described in a granted UK Patent [13].

B. Need for mobility

Although, the above mentioned simulator works well for people with access to a desktop or a laptop, it has a big constraint in terms of mobility. A person has to have access to a desktop or a laptop at most times to be able to achieve maximum success rates. There also lies the case of people with limited or no access to a computer.

Mobile devices such as mobile phones and PDA's provide a possibility of solving the issue of mobility. It is estimated that nine out of ten people within UK carry a mobile phone [14]. "Smartphones" (mobile phones with higher computational capabilities and resources), come with added facilities for wireless data transmission(Bluetooth) and mobile internet(WAP), also these feature larger memory and screen sizes. The increasing popularity of smartphones and increased functionality that such phones possess at a cheaper price, makes them a popular choice compared to the expensive PDA's.

Intel research laboratory [15], in collaboration with University of Calafornia, Berkley introduced the Mote technologies. "Motes" are small, self contained, battery powered computers with radio link enabled in them to allow exchange information between them. Intel plans to collaborate further with the research community to develop new wireless applications. One of the areas of interest has been wireless health monitoring.[16].

Together, these technologies have the possibility of providing a powerful wireless health monitoring network.

III. RESULTS

The hypothesis behind this project is that the components now exist to implement a radically different approach to healthcare, but that these technological components have yet to be integrated to form a fit for purpose system. These systems allow individual patients to benefit from a longer and higher quality of life, and allow healthcare professionals to become more effective and efficient in their care of patients. This project seeks to achieve this integration, and trial the result with patients and healthcare professionals. This project vision is articulated in Fig 2 and Fig 3. Fig 2, outlines the data collection routines used in conjunction with the smartphone. Fig 3, shows the possible integration of such a system with a health care centre. This is an evolution of our previous work[17][18].

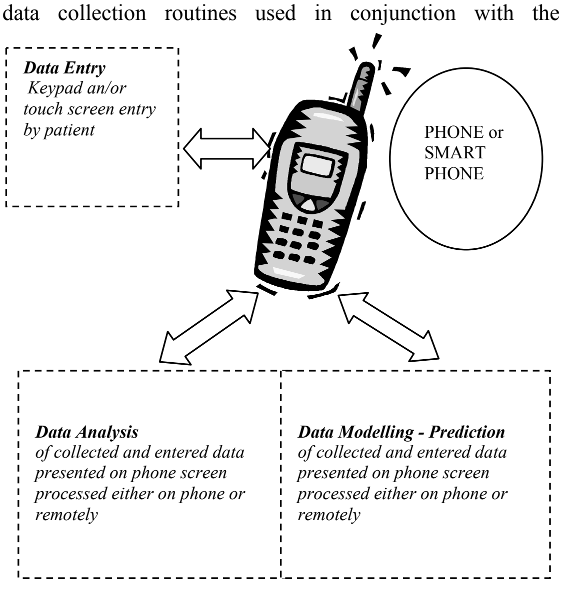


Fig 2.Data collection routines associated with the Smartphone

IBM research trials [19] have shown the possibility of using Bluetooth technology for wireless data transmission between mobile phones and specialised health monitoring sensors. As part of their trial, researchers at IBM, took specialised prototype blood pressure sensor integrated with Bluetooth capabilities and transmitted data onto a mobile phone. The phone acts as a transmission bridge, transferring the data to a health care professional. This approach confirms the ability of using Bluetooth as a medium of transferring vital health data, and using mobile phones as a go between the health care and the patient. However, the system does not record any more data regarding the patient's health such as their diet records, activity records. This additional information along with the blood pressure readings would allow better and more accurate modelling of the patient's health condition.

Data collection routines consist of a manual data entry of the individual's daily diet composition, activity listing, medication details and blood glucose readings. The system also provides other tools such as BMI calculator and Salt intake calculator. The proposed system will build upon prior work and experience in understanding, and developing software to meet, the needs of people with diabetes [11], [12]

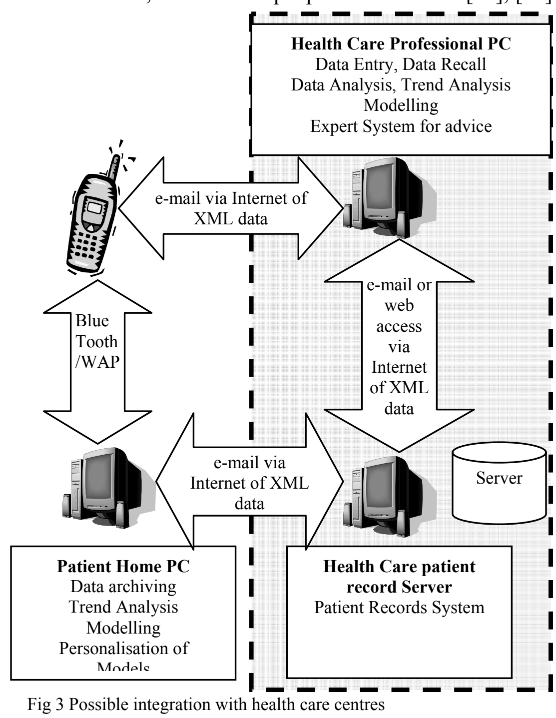


Fig 3 Possible integration with health care centres

is a key contribution in this field of developing fit for purpose, patient friendly, user interface for the acquisition of information about patients diet.

The novel contribution this project can offer lies in the collection, analysis and modelling of personal data acquired by wireless/wired sensors and/or by user entry. There exists a possibility for further contribution by researching proactive computer software systems to utilise these data to assist in the delivery of the e-health vision of the NHS Scotland IM&T strategy. The patented approach of Modelling of Metabolic Systems [13] will prove vitally important in the development of these proactive systems and the delivery of this vision.

A. Case example - people with type 1 diabetes, or type 2 diabetes taking insulin

The project aims to harness the power of mobile technology to give the user continual access, on the move, to near real time information regarding their health condition.

- Data is entered manually by the patient into the Smart phone application using the phone keypad and can include a past or present record, and future plan, of the patient's diet, activity details and medication taken, such as insulin doses, types and times of doses. The obtained data is modelled within the phone using a data modelling software. The patient can then choose to view the modelled data for different time intervals such as daily, monthly weekly or quarterly models as shown in Fig 4.

- Currently, the populated models are stored on the phone for a maximum of 90 days due to phone memory constraints. Patients can recall the modelled data any number of times and will also have the option to transfer these onto a computer using Bluetooth or an Http Connection. Future options may include a possible collaboration with health care institutes to which these data sets can be sent directly using WAP or as messages.

Small scale analysis and modelling applications available to the patients on their Smart Phones, or as services accessed via their phones, will empower them to do on the spot checks on their current health status. Results could then be sent to a web service, or the patients PC, for long term storage, amalgamation with prior data or for further analysis and modelling.

IV. Validation Trials

We are setting out plans to carry out validation trials for some of the tools used in our systems. Through these trials we wish to evaluate the following:

- To evaluate the accuracy of our system in comparison with a similar system done manually

- To compare the frequency of gathering health information using both technique and also the shortcomings in either systems

- To analyse the acceptability level of such a system by individuals for use in day to day life

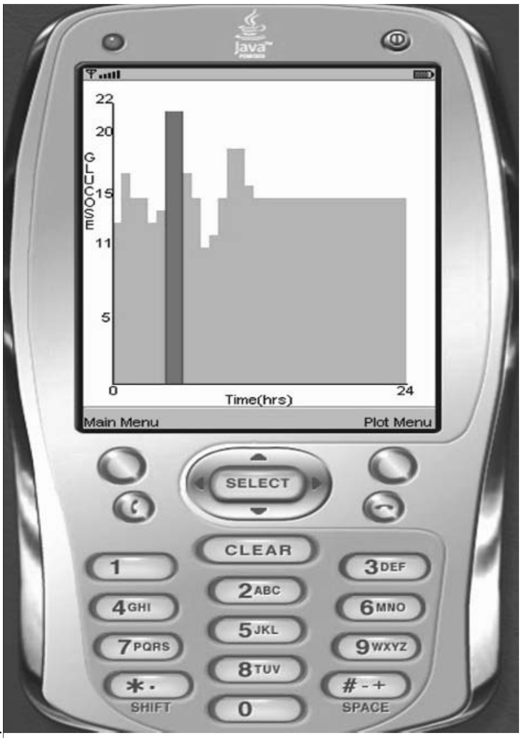


Fig 4 Modelled blood glucose

V ADVANTAGES AND LIMITATIONS

A. Advantages

- Increased frequency in monitoring individuals and recording their relevant health information.

- Faster health analysis and reporting compared to conventional hospital based health checkups.

This page intentionally blank

- Options for a patient or an associated health care professional to populate a patients health history as and when required.

- Future, possibility of integration with NHS health record systems.

B. Current limitations

- Computational capabilities : Medium range mobile phones still lack enough computational power to carry out complex computational work. J2ME a subset of Java programming language is designed for mobile phone application but is restricted in terms of development capabilities

- Memory: Having less memory compared to their desktop counterparts. This introduces new constraints in terms of database implementation or collection of records and storing of modelled images over a large period of time.

- Battery life : High computational processes consume system resources much quickly. Battery life plays an important factor in determining the complexity of the applications being developed.

- Security : There have been a few concerns with regards to wireless technology and with mobile viruses. Encryption and work from major anti-virus companies is at least a step in the right direction.

Application development is thereby seriously hindered because of the current technological limitations. However, with the rapid development of new technologies and the increased growth and development of Smart phone, it is possible to overcome these limitations in the future. Even with existing phones that are available in the market, this project has the potential to make a difference.

VI. CONCLUSIONS

A vision for the use of proactive mobile computing tools to empower people with chronic conditions has been presented, and work in progress towards attaining this vision has been described.

VII. REFERENCES

[1] http://www.who.int/diabetes/en/ accessed on 30th March 2005.

[2] http://www.who.int/hpr/NPH/docs/gs_obesity.pdf , accessed on 30th March 2005

[3] NHS acute sector expenditure for diabetes: the present, future, and excess in-patient cost of care. Currie CJ, *et al*. 1997. *Diabetic Medicine*, **14**: 686-692

[4] NHS Finance 2002-03: the issue explained, http://society.guardian.co.uk/publicfinances/story/0,12671,847924,00.html

[5] The effect of intensive treatment of diabetes on the development and progression of long-term complications in insulin-dependent diabetes mellitus. The Diabetes Control and Complications Trial Research Group, *N Engl J Med* 1993 Sep 30;329(14):977-86

[6] Prospective Diabetes Study (UKPDS). VIII. Study design, progress and performance. Diabetologia. 1991 Dec;34(12):877-90

[7] http://www.dh.gov.uk/PolicyAndGuidance/HealthAndSocialCareTopics/Diabetes/fs/en [last accessed 20th May 2005]

[8] http://www.scotland.gov.uk/library5/health/sdf-00.asp [last accessed 20th May 2005]

[9] http://www.show.scot.nhs.uk/imt/ , accessed on 30th March 2005.

[10] Scott A DR. Improving outcomes for young people with diabetes: use of new technology and a skills-based training approach is urgently needed. *Diabetic Medicine* 2001;**18**:861-3.

[11] Butler R,Gold A .E, Love J, Smail P, Stewart D, Usher J.R, “Evaluation And Refinement Of A Lifestyle Model For People With Type 1 Diabetes”, Decision Support in Diabetes Care, Eighth Workshop of the Diabetes care Optimization through Information Technology (DOIT) Study Group of the European Association for the Study of Diabetes (EASD),Conference Centre, Stirling Royal Infirmary, Stirling, United Kingdom, 7th - 8th September 2001.

[12] Franklin V.L. , Butler R, Wilson A, Greene S.A, “Validation Of A Computerised Diabetes Simulator Using Continuous Blood Glucose Monitoring In Children With Diabetes”, Poster Number **P106** Presented at Diabetes UK Annual professional Conference 2004**,** Birmingham, 17-19th March 2004. Published in Diabetic Medicine**:** April 2004. - Volume 21 Issue s2.

[13] UK Patent GB2381907 , “Modelling Metabolic Systems”, Inventor Dr Richard Butler, Applicants Univation Limited , Dr Richard Butler, Filed 6th September 2002, Granted 14th April 2004.

[14] http://www.bbc.co.uk/bob/mobilephones/ , accessed on 20th May 2005

[15] http://www.intel.com/research/exploratory/motes.htm, accessed on 20th May 2005.

[16] http://www.intel.com/pressroom/archive/releases/20040316corp.htm , accessed on 20th May 2005.

[17] Mathews, A. G., & Butler, R. (2005). *A Vision for the Use of Proactive Mobile Computing tools to Empower People with Chronic Conditions*. In the Proceedings of Eighteenth IEEE symposium on COMPUTER-BASED MEDICAL SYSTEMS, Dublin, 22-24th June 2005, pp. 425-427.

[18] Mathews, A. G., & Butler, R. (2005). *Mobile Computing Tools as a Medium to Empower and to Educate Individuals with Chronic Conditions*. In the Proceedings of The 6th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, Liverpool UK, 27-28 June 2005, pp. 483-487

[18] http://domino.research.ibm.com/comm/pr.nsf/pages/news.20031112_mobilehealth.html, accessed on 30th March 05

Design of a fast, low-level fault-tolerant protocol for Network on Chips

Muhammad Ali[1], Awais Adnan[2], Michael Welzl[1]
[1]Institute of Computer Science, University of Innsbruck, Austria
[2]Institute of Management Sciences, Peshawar, Pakistan

Abstract: *Network on a chip (NoC) has been proposed to address the inefficiency of buses in the current System on Chips (SoC). However as the chip scales, the probability of errors is also increasing, thus, making fault tolerance a key concern in scaling chips. Transient faults are becoming a major cause of errors in a packet based NoC. A transient error can either corrupt the header or the payload of packet requiring a retransmission from the source. Due to retransmissions, packets arrive out of order at the receiver side. Complex reordering algorithms are required at the receiver side to organize packets before sending them to associated resource. This adds a major overhead as the storage and logic capabilities are limited on a chip. We therefore provide a low-cost, fast and reliable end-to-end protocol for NoC which does not require reordering of the packets at the receiver end. Our protocol performs bitwise logical operations using binary representation of the addresses for the buffers to handle packets, hence making it much faster than conventional algorithms. Furthermore, the protocol is equally applicable to both static as well as dynamic routing environments on a chip.*

Keywords: Fault Tolerance, Network on a chip, Transient / soft errors.

I. INTRODUCTION

The International Technology Roadmap for Semiconductors (ITRS) projects that by the end of this decade chips would accommodate billions of transistors [1]. This means that in near future Application Specific Integrated Circuits (ASIC) will be made up of hundreds of communicating partners. However, it has been observed that the existing on-chip interconnects pose a serious threat toward achieving the billion transistor era. Buses that form an integral part of today's SoCs show serious degradation of performance beyond a certain number of communicating partners [2]. Considering the unmatched success of the Internet especially in terms of scalability, VLSI researchers have borrowed ideas from computer networks and proposed a packet based communication model for SoCs. This new paradigm is termed as *Network on Chips (NoC)* [3]. The idea is to send data in the form of packets over a network of switches/routers on a chip. This way communication and computation are kept transparent from each other, hereby achieving reusability of components and scalability of the whole network on-chip.

The increasing number of transistors on a chip is also contributing toward an increase in the faults, both temporary and permanent. With decreasing die size, cross talk, critical leakage currents and high field effects will increase the probability of permanent and temporary faults on a chip [4]. Permanent faults may cause one or more on-chip links or switches to fail, causing permanent damage to the component. Temporary faults, also known as transient failures or soft errors, do not cause permanent damage but may scramble one or more bits in a packet, hence making it invalid [5].

The frequency of transient errors is much higher than the permanent faults on-chip, making it necessary to provide built-in error recovery mechanism. With increasing transistor density on-chip, the soft error ratio is increasing exponentially [6]. Since today's SoCs are integrated into consumer products, it is important to equip them with some degree of fault tolerance. This can increase the overall yield of the chips by reducing the cost of production.

In this paper we present a fast and reliable end-to-end protocol for NoCs which ensures safe delivery of packets from source to destination. Since the protocol is implemented in the end systems, no complex logic algorithms and excessive storage capability is required at the intermediate routers. The significant aspect of our protocol is its simplicity in implementation on a chip as it uses bitwise logical operations to process and store the packets, eliminating any need for complex reordering and storing algorithms.

The paper is organized as follows; in the next section we discuss issues regarding error tolerance in NoCs along with related work. In section III, we provide a brief discussion of the NoC model we are using. Section IV elaborates the description of our protocol in detail, followed by conclusion and future work.

II. FAULT TOLERANCE IN NOCs

Traditionally, chips are equipped with error detection and correction mechanisms to deal with transient faults. However, packet based communication on-chip brings new challenges in terms of error resilience. The primary unit of packet based communication is a packet itself. A packet is usually composed of a *header* and a *payload*. The header of the packet contains identification information like source and destination addresses, routing path, CRC (Cyclic Redundancy Check) checks etc. The payload contains the actual data to be transported. A soft error can scramble either header or the payload of a packet. A bit flip, due to a soft error, in the header of the packet may cause it to be routed to a wrong destination. An error in the payload, on the other hand, may corrupt the contents of the packet, hence making it invalid although it might reach the destination. In both cases a retransmission of the misrouted or corrupt packet is required.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 345–348.

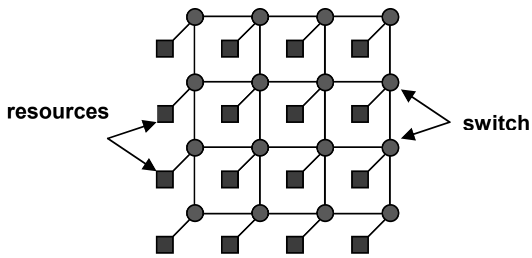


Figure 1: 2D mesh topology for NoCs

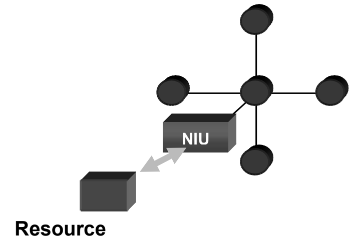


Figure 2: Abstract representation of NI

The authors of [7] have discussed and analyzed various error coding schemes for NoCs. In [8], T. Dumitras et al. present a stochastic communication model for NoCs based upon rumor spreading. In this mechanism a node spreads the packet to its neighbor which, in turn, spreads it to all its neighbors assuming that at least one packet will reach the destination. Though simple, the mechanism has high packet overhead especially when a large number of nodes are communicating. In [9], Bertozzi et al. present a link level flit[1] based retransmission protocol. Each flit is acknowledged by every intermediate router on successful reception. In case a flit is missing or is corrupt, a negative acknowledgement is generated. Such a mechanism adds complexity on part of intermediate routers besides buffering them.

We propose a fast, efficient, and fault tolerant scheme to deal with packet corruption due to transient errors in a NoC. The mechanism does not require intermediate routers to maintain buffers and thus forwards packets as they arrive. Moreover, the mechanism is equally effective in both dynamic and static routing environments of NoCs. The detailed description of the proposed protocol is discussed in section IV.

III. NOC MODEL

Various topologies are possible for NoCs like honeycomb [10], fat-tree [11], 2D mesh [12] etc. The choice of a topology can dramatically affect the network characteristics of a NoC in terms of number of hops, link delays etc. Although we intend to study the behavior of our mechanism with other topologies in future work, in this paper, we only discuss the most agreed upon topology for the sake of simplicity --- a *2D mesh*. A typical topology model is shown in figure 1 followed by its description:

Boxes represent resources in a NoC which is an end system having data to send or receive. Each resource in this scenario is connected to a router. They are considered black boxes as they can be anything: a MIPS[2] processor, a DSP[3], a memory module or even a combination of any or all of these elements. Hence in terms of a set of resources, it can be termed as *network within a network.*

Circles in figure 1 represent routers which are responsible for establishing a routing path between sender and receiver. Each router is directly connected to neighboring four routers (except for the ones at the edges), thus creating a mesh network.

Network Interface (NI) is a special device acting as a middle layer placed between the resource and the router. NI is responsible for creating packets from bit streams obtained from the connected resource and vice versa. It is a very significant unit of the NoC as it covers the discrepancies that exist between the homogeneous router network and different kind of resources emanating from a variety of vendors. NI makes it possible to reuse not only the resources but also the network architecture of a NoC. A typical example of a NI is shown in figure- 2.

IV. DESCRIPTION OF THE PROTOCOL

As mentioned earlier, we propose an end-to-end reliable packet delivery mechanism, restricting the intricacies of the protocol to the sender and receiver nodes. The sender sends a pre-defined number of packets to the receiver and after making sure their safe delivery, sends the next set. If there are X packets in a set, it means the sender and receiver need X buffers to store them. In our example setup, we take it as 16, which means the sender and receiver have to buffer 16 packets at a single time. Each packet in a set is identified by a unique packet ID from 0 to 15 (*0000 -1111 at lower four bits of address*). The receiver associates a flag value with each buffer which is initially *zero* . When a packet arrives at the receiver, the error detection code checks it for any inconsistencies (scrambled bits, for example). If a packet arrives with incorrect payload, it is dropped. The receiver sets the flag for each received packet as 1 and otherwise for packets not received.

Consider figure 4 which is an abstract representation of block diagram at the receiver side. The packet ID is first ANDed with equal number of bits where except for first 4 bits all the rest are zeros. This way, we can retrieve the

[1] Packet divided into equal sized smaller units called *flits*
[2] *M*icroprocessor without *I*nterlocked *P*ipeline *S*tages
[3] *D*igital *S*ignal *P*rocessor

Table 1: showing packet numbers, their binary equivalent and corresponding 16 bit register values

Packet ID (p)	Packet ID (4 bits)	*T*= 2 ^ p	In 16 bits
0	0000	1	00000000 00000001
1	0001	2	00000000 00000010
2	0010	4	00000000 00000100
3	0011	8	00000000 00001000
4	0100	16	00000000 00010000
5	0101	32	00000000 00100000
6	0110	64	00000000 01000000
7	0111	128	00000000 10000000
8	1000	256	00000001 00000000
9	1001	512	00000010 00000000
10	1010	1024	00000100 00000000
11	1011	2048	00001000 00000000
12	1100	4096	00010000 00000000
13	1101	8192	00100000 00000000
14	1110	16384	01000000 00000000
15	1111	32768	10000000 00000000

first four bits of the packet ID which is unique for each packet in a set[4]. The packet number P is converted into 2^P by a temporary register, ***T,*** which is done by shifting 1 *p* times to the left. For instance, if packet 5 arrives, its packet ID is 4 (since the count is from 0 to 15); 2^4 gives 16 (10000) which is achieved by shifting 1 five times to the left of 16 bit register yielding 0000 0000 0001 0000. Similarly, for every different packet 1 has a uniquely significant position in 16 bits register. Table 1 shows the values for ***T*** and 16 bit register for all 16 packets in a single set. In order to detect any duplication in received packets, the temporary register ***T*** is first NANDed with the 16 bit register. If it yields a zero, it means the packet is a duplicate and is dropped, otherwise, the packet is accepted. These logic operations are primitive in nature and can be implemented simply on a chip. Figure 4 is an abstract illustration where in actual they are a group of 16 similar gates one for each bit.

To store packets directly in their respective locations, we consider the buffer as a continuous storage of size *Z=N*K,* where *N* is the number of packets (16 in our example), and *K* is the size of each packet (say 32 bits)[5]. Logically speaking, the buffer is divided into *N* sub-buffers each of size *K.* The starting address of each packet can be calculated by multiplying first four bits of its unique packet ID to the packet size. This can be achieved by shifting 5 bits left of the 4-bit packet ID. For example, to find out the location for 5th packet in the buffer, we multiply its packet ID (4) with packet size 32, which yields 128 = 010000000. This binary value is obtained by shifting 5 bits to the left of packet ID

A. Retransmission policy:

The next important stage is for the receiver to let the sender know if it received all the packets or not. In case it received all the packets, it has to acknowledge the sender so that the sender can send the next set of packets. In case one or more packets were dropped, they need to be retransmitted before the buffers are relinquished. In the given scenario, there could be two methods to request for retransmissions which are based upon the routing strategy implemented.

A.1. Static routing

In static routing, paths are fixed and communication is carried out following the shortest path. One example of static routing is XY based routing where a packet is first routed in the X direction and when it comes in the column of the destination, it moves in the Y direction until it reaches the destination [12]. Since deterministic routing always takes the shortest possible path, it is likely that all the packets follow a particular path. In such a case, all the packets are supposed to reach in order at the receiver side. When a corrupt packet reaches the receiver, the receiver drops that packet and generates a negative acknowledgement for the retransmission of the packet. In such a situation, using our mechanism, the receiver will receive the packets as they come in and will put them in their respective buffers. Every time a packet is dropped either by the receiver or at some intermediate router, the receiver issues a negative acknowledgment to the sender. The sender, on receiving the negative acknowledgement resends the requested packet. Since our protocol uses bitwise addressing with shift registers to place the packet in their exact locations, no reordering is required. Once all the packets are received successfully, the receiver sends a positive acknowledgment upon which the sender relinquishes the buffers and fills them up with the next set of data. In case that a requested packet does not arrive, the receiver issues a second negative acknowledgement after a certain timeout[6].

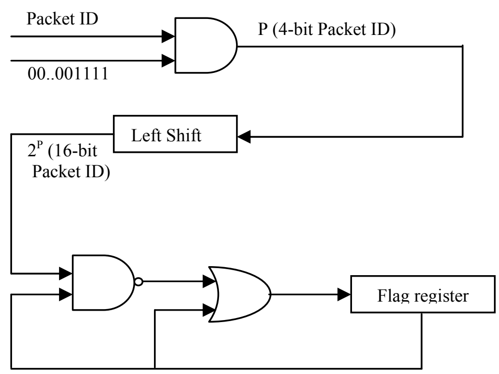


Figure 4: A block diagram for receiver

A.2. Dynamic routing

Dynamic routing, as the name shows, allows to change the routing path of packets in a NoC automatically in case a router fails or congestion occurs in the optimal path. In dynamic routing packets may take different paths to reach the

[4] Assuming that each set is of 16 packets, each packet in a set will have unique ID.
[5] This value is only for simplicity since the actual packet size can range from 32 to 64 bytes.

[6] timeout value depends upon the routing strategy implemented.

destination. In this case, they may not arrive in order at the destination since different paths may have different latencies. Dynamic routing is more flexible in terms of congestion in the links or permanent failures; however, it is more complex in terms of implementation. Each router needs to be equipped with dynamic routing protocol to deal with changing situation in the NoC.

The retransmission policy in terms of dynamic routing is as follows; the receiver receives all the packets according to the method explained above. Then after a certain timeout, it scans all the flag registers and for those registers which are still zero are sent to the sender which resends all those packets. Alternatively, the receiver can simply send the flag registers to the sender which then scans them and resends all those packets for which the register value is still high. In either case, the missing packets are delivered to destination.

V. CONCLUSION AND FUTURE WORK

In this paper we have proposed an efficient, fast and reliable packet delivery protocol based in the end systems. The protocol entirely uses bitwise logic making it simple to implement on a chip. Buffers are only kept at the source and destination. Since the protocol uses registers and logical operations to store packets in their respective locations, therefore, no reordering algorithms are required. Considering the limited storage and logic capabilities on chip, our protocol proves to be cost effective besides providing reliability in the presence of soft errors.

We are currently working to implement our reliable protocol for NoCs in Network Simulator NS-2 [13]. The simulation setup is considered for a 4x4 2D mesh based NoC. The die size is assumed to be $22mm^2$ with a link size

Table 2: shows addresses for packet numbers from 0 to 15

Packet ID	4 Bit Packet ID	Packet Size K	Starting Address K_n=n*K	In binary format
0	0000	32	0	0000 00000
1	0001	32	32	0001 00000
. . .	. . .	. . .	. . .	. . .
14	1110	32	448	1110 00000
15	1111	32	480	1111 00000

of 4.5mm each. The capacity of each link is 2Gbits/sec with a link delay of 10 nano-secs. Furthermore, we propose a 4x4 cross bar router architecture. Each output may be connected to at most one input, whereas each input may be connected to as many outputs as possible. A crossbar router is critical in case of dynamic routing where a packet may be forwarded on any output channel due to network load or physical conditions.

At the moment we are only concentrating on unicast[7] routing. Multicasting[8], on the other hand, is a very complex issue and involves many parameters of a NoC to be reconsidered. For instance, if a source is communicating with many destinations, the key issue is about the communication paths; weather the same packet be sent via different paths or via same path especially when the destinations are situated at the same level. Another possibility is to send a single copy of packet until it reaches one router before the first destination. This router then replicates the packet and sends to other attached destinations. Also in a fault tolerant environment, the sender needs to keep track of packet corruption or losses for each individual receiver. All such issues are quite complex in nature but nevertheless worth looking at. We thereby are determined to implement and test various multicasting mechanisms for NoCs in our future work.

[7] single source sending packets to a single receiver

REFERENCES

[1] www.itrs.net

[2] Ahmed Jerraya, Hannu Tenhunen and Wayne Wolf, ``Multiprocessor System-on-chips'', *IEEE Computer magazine*, July 2005, Vol. 38, No. 7

[3] L. Benini and G. De Michelli, `Networks on Chip: A new paradigm for component-based MPSoC design'', *Multi-processors Systems on Chips*, edited by A. Jerraya and W. Wolf, Morgan Kaufman, 2004, pp. 49-80.

[4] Ming Shae Wu and Chung Len Lee, ``Using a Periodic Square Wave Test Signal to Detect Cross Talk Faults'', *IEEE Design & Test of Computers*, Volume 22, Issue 2, March-April 2005, Page(s): 160 - 169

[5] Michael L. Bushnell, Vishwani D. Agarwal, ``Essentials of Electronic Testing for Digital, Memory and Mixed-Signal Circuits'', *Kluwer Academic Publishers*, USA, 2000

[6] P. Shivakumar, M. Kistler et al. ``Modeling the effect of technology trends on the soft error rate of combinational logic'', Proc. DSN 2002, pp.389-398

[7] Srinivas Murali, et al. ``Analysis of Error Recovery Schemes for Networks on Chips'', IEEE Design & Test, Vol. 22, Issue 5, Sept. 2005

[8] T. Dumitras, R. Marculescu, ``On-Chip Stochastic Communication'', *Proceedings, DATE 2003*, pp. 790-795

[9] D. Bertozzi et al., ``Xpipes: A Network-on-chip Architecture for Gigascale System-on-chip'', *IEEE Circuits and Systems*, 2nd quarter, 2004, pp. 18-21

[10] Ahmed Hemani, Axel Jantsch, Shashi Kumar, Adam Postula, Johny Oberg, Mikael Millberg, Dan Lindqvist, ``Networks on a chip: An Architecture for billion transistor era'', *Proc. IEEE NorChip Conference*, November 2000

[11] P.Guerrier and A.Greiner ``A generic architecture for on-chip packet switched interconnections´´, *Proceeding Design and test Europe,* March 2000

[12] William J. Dally and Brian Towles, "Route packets, not wires: on-chip interconnection networks, Proceedings, *Design Automation Conference (DAC)*, pp. 684-689, Las Vegas, NV, June 2001

[13] http://www.isi.edu/nsnam/ns/

[8] single source communicating sending packets to many receivers

Leakage Power Proliferation in Short Channel Cache Memories

N. Mohamed and N. Botros
Department of Electrical and Computer Engineering
Southern Illinois University, Carbondale, IL 62901-6603

***Abstract*-This work investigates the escalation of leakage power loss in today's low power processors as their feature size shrinks rapidly. The problem seems to exacerbate as technology scaling dives steadily into Very Deep Submicron (VDSM). A quantitatively analysis has been carried out in several cache systems and has shown that the 1-way set associativity optimizes this power component across various generations.**

I. INTRODUCTION

Up till recently, the leakage dissipation in most microprocessors-the one that used to be negligible- has increasingly become a dominant factor. With well above 40% of the StrongARM power being lost in cache system [1], cache dissipation is widely considered to be reflexive of microprocessor power loss. This work focuses on the leakage component of that loss. StrongARM, a commercially well-known processor, has been selected to represent a class of low power embedded microprocessors. It has been classified into several variants with each meant to represent an alien of a scaled down generation. For each variant, the impact of the leakage current contribution to cache total power loss has been studied quantitatively. The effect of cache organizations in each generation has also been scrutinized. The resulting performance degradation associated with that has been investigated as well. The rest of this paper is organized as follows: section II presents the basic building block of standard SRAM and the models that are used in simulation. Section III studies the impact of the reduced cache feature size on leakage current dissipation. Here, we show that cache organization can be used to boost the associated performance loss. In section IV, we present our experimental results and conclude in section V.

II. LEAKAGE POWER MODELS

Ideally, SRAM arrays dissipate no static power. Yet, as these arrays start to downsize to meet the growing demand for low-power processors, they have increasingly become leaky. The main reason for that is the reduction in CMOS inverter threshold voltages- known as threshold roll off- that is associated with the supply voltage reduction. This threshold roll off marks a trade off between power and speed at the device level of abstraction. For while reducing device supply voltage does drastically reduce its power dissipation, as well as its speed, the latter effect is normally offset by cutting on threshold voltage [2]. Hence the device becomes more vulnerable to standby current when it is presumed turned-off. This emerging current component does not seem to show any sign of abating as technology dives steadily into VDSM. In contrary, it has become the dominant factor [2]. The major component of leakage current is known as subthreshold current; I_s . It flows from drain to source due to the diffusion of minority carries when the NMOS device operates in the weak inversion region as shown in figure 1. This is in contrast to the drift current which dominates when the device operates in the strong inversion region.

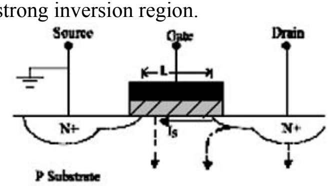


Figure 1: Leakage current components in NMOS transistor

The magnitude of I_s is given by [4]:

$$I_{sn} = (W/L)\ \mu\ V_{th}^2\ C_{sth} \exp[(V_{gs} - V_T + \eta\ V_{ds})/n\ V_t][(1-\exp(V_{ds}/V_{th}))] \quad (1)$$

Where W and L are the device width and length, μ is the carrier mobility, $V_{th} = k\,T/q$ is the thermal voltage at temperature T, C_{sth} summation of depletion region capacitance and interface trap capacitance per unit area of the gate, η is the drain-induced barrier lowering(DIBL) coefficient and n is the subthreshold slope shape factor given by:
$n = 1 + C_{sth} / C_{ox}$ where C_{ox} is the gate input capacitance per unit area of the gate. The same relationships apply for the PMOS transistors as well.

For SRAM-based caches that are built using such transistors, like that of figure 2, a model is required to calculate the cell total

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 349–352.

leakage. In terms of the Data Retention Voltage (DRV), Qin el al. [5] provides an accurate relationship where the overall leakage current I_s is calculated as:

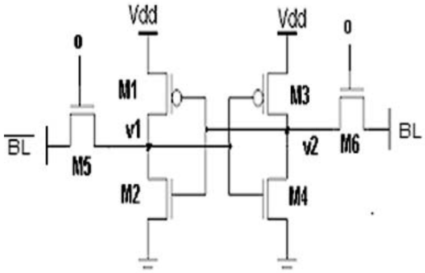


Figure 2: Six-transistor CMOS SRAM cell

$$I_s = I_{s2} + I_{s4} \quad (2)$$
$= (W/L)_2\ i_2 \exp[v_2\, q/n_2kT)].[\ 1 - \exp(-v_1q/kT)]\ +\ (W/L)_4\ i_4 \exp[v_1q/n_1KT].[1\text{-}\exp((v_2\text{-}DRV)q/kT)]$
where
$$i = I_0 \exp[-V_{th}/ (n\ k\ T/q)] \quad (3)$$
and I_0 is given by:
$$I_0 = \mu_o C_{ox}(W/L) V_{th} \exp(1.8) \quad (4)$$
with μ_o being the zero bias mobility.
$v_1 = (k\ T\ (A1 + A_5)/q\ A_2) \exp[-q\ DRV_1/n_2\ k\ T]$ $v_2 = DRV_1 - kT\ .A_4/q.A_3. \exp[-q\ DRV_1/n_3\ kT]$ and
$A_i = I_0\ (W/L)_i \exp(\ -q\ V_{th}/k\ T\ n_i\ )$

These models reveal an asymmetry in the basic cache cell structure, as well as in transistor characteristics and physical geometry. Integrating these models into the power estimator, have led to more accountable results. For various power estimations, this work employed a modified version of cacti [6], and Wattch [7].

III. SHORT CHANNEL LENGTH CACHE IMPACT

The equations of the precedent section clearly show that the leakage power dissipation in CMOS devices is a strong function of transistor channel length (L). While shrinking (L) decreases the gate capacitance of the device and hence its dynamic power consumption [3], it stimulates, on the other hand, the standby current component leading to an increase in leakage dissipation. Moreover, such an increase exacerbates as the threshold voltage is scaled down along with supply voltage to yield some performance boost. Voltage scaling is perhaps the most effective method of saving power due to the square law dependency of digital circuit active power on supply voltage. This trend is rampant in most modern process scaling. The associated leakage escalation is said to be due Short Channel Effect (SCE).For analytical purposes, we propose four models of the original StrongARM processor. Namely: Strong-1, Stong-2, Strong-3 and Strong-4 with Strong-1 being the baseline-fabricated with 0.18um process. These models represent the past, today and feature technologies. The leakage losses in each of these processor caches are investigated and analyzed independently.

TABLE-I
StrongARM VARIANTS AND CHARACTERIZATION

CPU Alias	Strong-1	Strong-2	Strong-3	Strong-4
Generation (um)	0.18	0.13	0.1	0.07
Supply Voltage(V)	2	1.1	1.2	1
Threshold Voltage(V)	0.2	0.1	0.09	0.001

The unwanted effect of shortening device channel length comes at the expense of great performance degradation. The problem has been tackled at a higher level of design abstraction by some research. Khouri et. al. [3] has dealt with the problem at the gate level and provided great optimization. We approached the problem at higher level of abstraction exploring the cache microarchitecture. The cache system organization is targeted. The impact of cache associativity and cache-line size in the magnitude of the subthreshold current is examined. The instruction and data caches of the various StrongARM clones are modeled and run under different associativities. The cache sizes for all processors are kept the same as that of the baseline. The behavior of the corresponding subthreshold currents due cache reorganization is closely monitored. The results are presented thoroughly in the next section.

IV. EXPERIMENTAL RESULTS

All results have shown consistent reduction of total cache power when migrating from one generation to another as technology downsizes. The leakage loss echoed the same trend

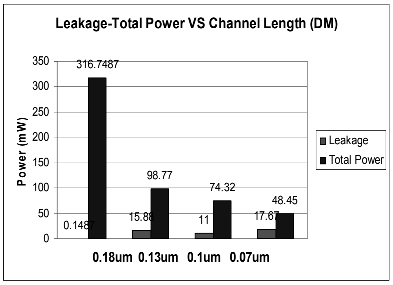


Figure 3: Cache System Profiles of Storng-1, Strong-2, Strong-3 and Strong-4 (when all directly mapped)

for all StrongARM clones. Increasing associativity and cache-line size, on the other hand, have led to gradual increase in leakage power for both instruction and data caches. This trend is depicted in figures 3 through 8.

The 32-set associativity of Strong-4 cache modeled with 0.07um technology, consumed the maximum leakage power while the directly-mapped of Strong-1 modeled with 0.18um consumed the minimum. The gap stretches for well above 500 m W. This huge proliferation of leakage among technology generations is bold between 0.18um and 0.13um as well as between 0.1um and 0.07um across all cache organizations. Yet, it is less obvious as we evolve from 0.13um to 0.07um where it momentarily decays. This is mainly attributed to the relatively higher supply voltage driving the latter generation with respect to the former – as shown in TABLE-I. The influence of supply voltage on leakage current has quadratic effect while channel length shrinkage impacts leakage linearly.

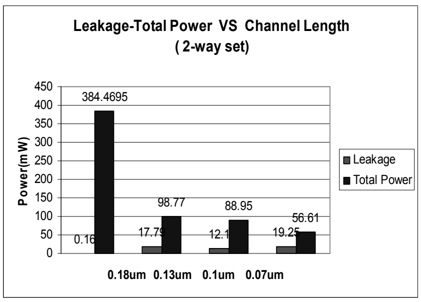


Figure 4: Cache System Power Profiles of Storng-1, Strong-2, Strong-3 and Strong-4 (when all 2-way set associative).

The ratio of leakage- to- total power was found to be minimum- 0.047 % - in 0.18um generation with 1-set associative while it was maximum -43%- in the 0.07um when associativity was set to 16. The disparity is clearly an indicative of growing standby power in future technology generations. As expected, 0.07um has shown the worse overall leakage proliferation. In this generation, with 32%, the 2-way associativity provided the minimum leakage-to-total power loss and therefore seemed favorable.

The results mark clear trade off between performance and power loss to leakage. Increasing associativity minimizes conflict miss rate in caches and therefore leads to an edge in performance. This

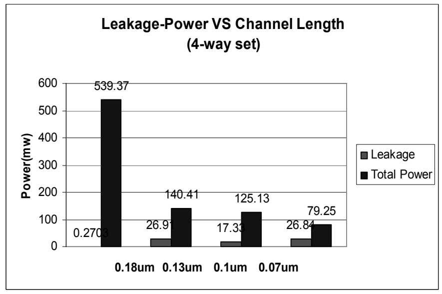


Figure 5: Cache System Power Profiles of Storng-1, Strong-2, Strong-3 and Strong-4 (when all 4-way set associative)

has been proved quantitatively in [9]. The performance enhancement comes at the price of more dissipation not only in dynamic component, but with an increased in leakage dissipation as well.

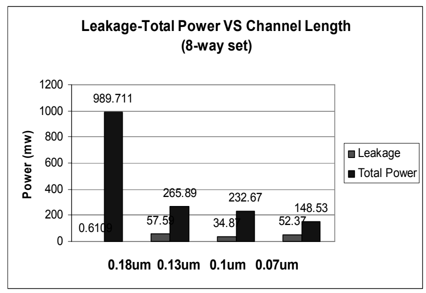


Figure 6: Cache System Power Profiles of Storng-1, Strong-2, Strong-3 and Strong-4 (when all 8-way set associative)

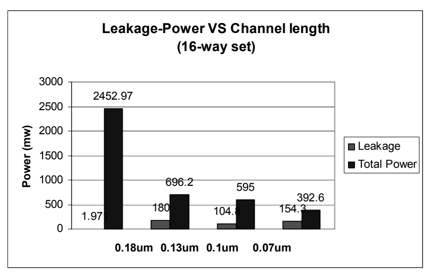


Figure 7: Cache System Power Profiles of Storng-1, Strong-2, Strong-3 and Strong-4(when all 16-way set associative)

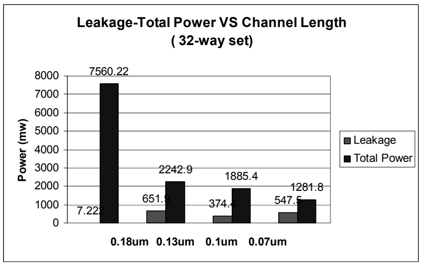


Figure 8: Cache System Power Profiles of Storng-1, Strong-2, Strong-3 and Strong-4 (when all 32-way set associative)

V. CONCLUSION

In this work, we have investigated the leakage growth in modern microprocessor caches as feature size increasingly shrikes. The impact of channel length demise in magnifying leakage current has been quantified and analyzed. Cache organization has been used to offset performance loss due technology migration. The 1-way set associativity yielded the optimal leakage current and seemed to be a favorite choice across generations. An embedded microprocessor has been used for the analysis. The results depicted where we stand today with respect to leakage and how vulnerable are modern processors to this kind of power dissipation.

REFERENCES

[1] J. Mantanaro et al. " A 160- MHz, 32b , 0.5-W CMOS RISC Microprocessor ", IEEE JSSC, pages 1703-1712, Nov. 1996

[2] J. Rabaey, A. Chandrakasan and B. Nikolic "Digital Integrated Circuits: A design Perspective" 2nd Edition, Prentice-Hall, 2002

[3] K. Khouri and N. Jha "Leakage Power Analysis and Reduction during Behavioral Synthesis", IEEE Transactions on VLSI systems, Vol.10 No. 6, Dec 2002

[4] F. Fallah and M. Pedram, "Standby and Active Leakage Current Control and Minimization in CMOS VLSI circuits", IEICE Trans. Electron. VOL.E88-C, NO.4 April 2005

[5] H. Qin, Y. Cao, D. Markovic, A. Vladimirescu, and J. Rabaey, "SRAM Leakage Suppression by Minimizing Standby Supply Voltage," IEEE *International Symposium on Quality Electronic Design,* March 2004

[6] Reinman and N. Jouppi. CACTI 2.0: An Integrated Cache Timing, Power and Area Model WRL Research Report 2001/2, August 2001

[7] D. Brook et al, "Wattch: A frame work for architectural – level power analysis and optimization", Proc. Of ACM ISCA-27, Vancouver, BC, June 2000

[8] N. Mohamed, N. Botros and W. Zhang "The Impact of Cache Organization in Optimizing Microprocessor Power Consumption", Proc. of the 2006 International Conference in Computer Design, Las Vegas, June 2006

An e-Science Environment for Aerospace Applications on Teragrid

Nam Gyu Kim[¶], Jung-Hyun Cho[§], Yoonhee Kim[§], Chongam Kim[ʄ], Kum Won Cho[¶], Sang Boem Lim[¶]
[¶]Korea Institute of Science and Technology Information
Daejoen, Republic of Korea
[§] Dept. of Computer Science, Sookmyung Women's University
Seoul, Republic of Korea
[ʄ]School of Mechanical and Aerospace Engineering, Seoul National University
Seoul, Republic of Korea

***Abstract* - e-Science Aerospace Integrated Research System(e-AIRS) is one of e-Science projects in KOREA. This system has been developed for aerospace researchers to be offered total research environment which is integrated and collaborative environment to enable computational simulation and remote experiments via portal. This paper presents that the core part of this system is adapted to Teragrid at NCSA. By way of this goal components of e-AIRS system are updated and customized to Teragrd and is added existing grid portal technology. Also this system is interlocked with NCSA tools to use Teragrid resources. Through these efforts e-AIRS offers easy-to-use research environment to aerodynamic researchers on Teragrid.**

I. Introduction

Today many countries have research activities related to e-Science as a next research infra-structure such as UK e-Science and Cyberinfrastructure in US [1] [2]. These projects aim for providing application scientists to easily access and use the research environment such as massive computational and storage resources, and collaborative environment.

In Korea, Korean Institute of Science & Technology Information (KISTI) has been initiating a project, funded by the Ministry of Science and Technology, called "Korea e-Science Project" since year 2005. Main goal of this project is to build a Korea e-Science infrastructure. To achieve this goal, we lunched 5 initiative projects – an equipment project, an aerospace project, a nano-technology project, a biology project and a meteorology project.

Among these projects, this paper will discuss an aerospace system called e-AIRS [3]. This system offers aerospace researchers to share massive computational resources, a visualization tool, a collaborative method and experimental equipment like a wind-tunnel through a web portal. So this provides them to remotely access aerospace vehicle design data, the results of CFD (Computational Fluid-Dynamics) model execution or wind-tunnel testing and collaborate with results. Hence this makes them to reduce theirs efforts and avoid overlapping investment to share data of other researchers. Also this offers one-stop service to them involved from the beginning of design of aerospace vehicle, computation to visualization by step-by-step through a user-friendly interface on a portal.

We already present e-AIRS system in other papers [4] [5]. Main issues of this paper are to present how e-AIRS system is successfully adapted to Teragrid based on international collaboration between KISTI and NCSA [6]. Initial e-AIRS system was not developed Tegragrid in mind. This led us to customize and update e-AIRS system to fit in Teragrid and to interlock with other grid and Teragrid tools.

This paper is organized as follow. Section 2 gives the background of e-AIRS project and the approach for e-AIRS to adapt for Teragrid. Section 3~6 give the overview of the design of this system and the description for components: e-AIRS middleware, job submission and job monitoring with Clumon and customized portlets [7]. Section 7 summaries what has been achieved so far in this project and outlines the future work.

II. Background

A. e-AIRS

The e-AIRS, one of projects of the construction of national e-Science in Korea, aims at to establish the powerful and ease-use research environment to aerodynamic researchers. The focus of the project is collaboration of results between numerical wind- tunnel and a wind-tunnel experiment via a numerical wind-tunnel, remote wind tunnel experiment and visualization. Thus, as researchers actively study their research on one system, the goal of this is to provide aerodynamic researchers to the research environment customized to aerospace domain which is multi-disciplinary study made of Aerodynamics, Structure, Propulsion and Control. By way of this goal the e-AIRS system is made of 3 parts as in Fig. 1.

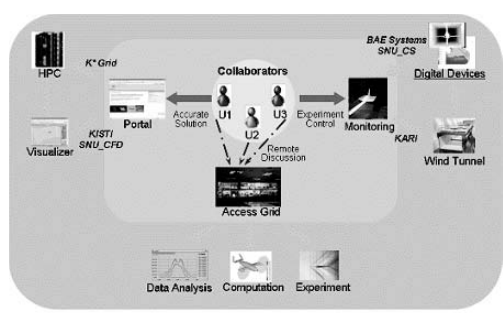


Fig. 1 System relation in e-AIRS

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 353–357.

The first part of this is numerical wind-tunnel service which consists of e-AIRSMesh, CFD solver service, data management service, e-AIRSView as a visualization tool. The e-AIRSMesh implemented by java applet as pre-processor is the mesh generator to generate geometries for aerospace vehicle as input data of CFD solver. CFD solver service is the portlet to select mesh data files and computation resources, submit a job to remote machines, transfer these data files for CFD simulation from the cluster to the storage and check process of calculation by graph. The e-AIRSView implemented by java applet as post-processor is the visualization tool to display the result of calculation as 3D image.

The second part is the remote wind tunnel service which remotely request wind tunnel experiment and receive the result of experiment through the portal. After the user requests wind-tunnel experiment on the portal, the operator who actually tests in wind-tunnel can check requested experiment on the portal. After the operator tests this and uploads result images to the portal, the user can view the result on this.

The third part is Access Grid which cooperates between researchers via video chat and sharing visualization.

This system is released for the first time in Jan 2006 and numerical wind-tunnel service and remote wind-tunnel service is serviced on web.

B. The Approach to the e-AIRS adapted to Teragrid

The e-AIRS is the total solution as PSE (Problem Solving Environment) for numerical wind-tunnel. And it has also the remote wind tunnel service and the Access Grid to collaborate between researchers.

As Fig.2 shows, flow of wind tunnel simulation service can be distinguished into three major parts which is pre-process, calculation, post-process. Pre-process will produce mesh by using mesh generator (we called e-AIRSMesh). We also can accept pre-existing mesh data. As a input format we can use raw CAD data which is accepted as a regular mesh input and is made suitable format for e-AIRSMesh. After mesh data is produced by e-AIRSMesh, the simulation part will perform simulation of CFD. CFD simulation will be distributed in many computational resources to get a best performance. Post-process can make view of simulation result using e-AIRSView as a visualization tool in 3D format.

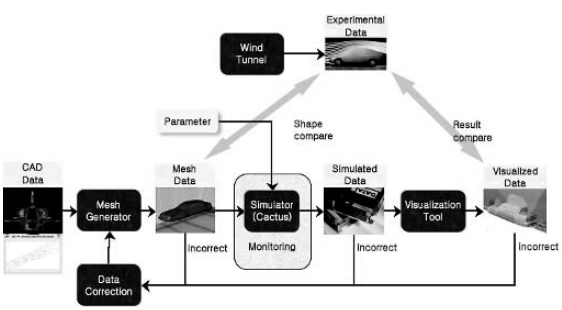


Fig. 2 Process of CFD on e-AIRS

However the e-AIRS System is alpha version, so some components are not perfect or tightly coupled to them or restricted by hardware resources. And the project, International collaboration between KISTI and NCSA, was given only about 8weeks. So we decided that only CFD solver service essential part of whole process is adapted to Teragrid. For this, we used and customized Grid Portlets on Gridsphere to use a grid credential management, a job submission and Clumon to monitor PBS job on Teragrid at NCSA [8]. And we used CFD solver made by Dr. Byoung-Do Kim who works at NCSA.

III. The Implementation of the e-AIRS adapted to Teragrid

A. The Architectural change of the e-AIRS for Teragrid

As Fig. 3 shows, the architecture of original e-AIRS system is changed to make suitable for Teragrid at NCSA. CFD solver service of e-AIRS System consists of e-AIRS middleware which has job submission, automatic result data file transfer, data management of jobs and files and portlets which are job submission UI, job monitoring, resource management, display of residual and time survey graph. This has only one grid credential to submit job because it manage and access all data by means of user management based on database and job runs only one cluster based on Globus without a job scheduler [9].

However, Teragrid at NCSA has PBS as a scheduler on to job submission and Clumon as the job & resource monitoring tool related with this PBS. And Teragrid is based on the grid credential by each user.

So among these functions of CFD solver service, job submission parts of e-AIRS middleware and a portlet are removed. Also job monitoring and resource management portlets are removed. Instead of these we used and customized Grid Portlets to use grid credential for each user, submit job and manage resources and Clumon as a job monitoring tool because all job is submitted and managed by PBS on Teragrid at NCSA.

On the other hand, 3rd party transfer via GridFTP and remote file monitoring about several result files produced by CFD solver are added to automatic result data transfer. This is the overall system diagram of Ge-AIRS System adapted to Teragrid at NCSA.

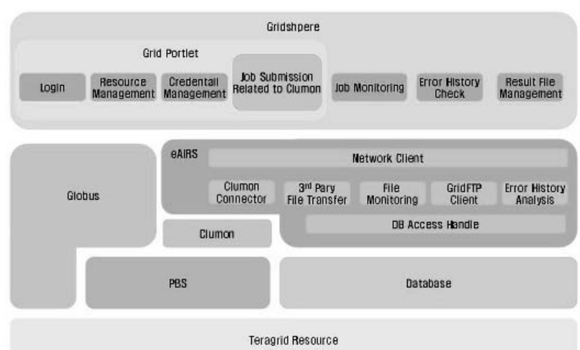


Fig. 3 System Architecture of e-AIRS adapted to Teragrid at NCSA

B. The relation of physical hardware resources

The whole physical hardware resource is showed in Fig. 4.

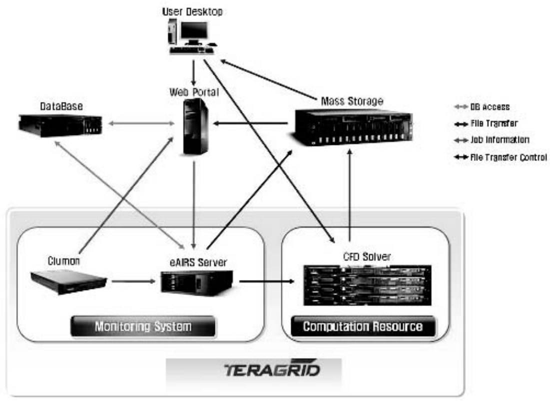


Fig. 4 Relation Diagram of Physical Hardware

The web portal server contains Gridsphere and Grid Portlets on Apache Tomcat to provide user UI on web to login the portal, manage a credential, submit a job, monitor a job, view result data list, display residual and time survey graphs and request download from the mass storage to user's own computer [10]. And the database server is stored job information and metadata of files which are produced by CFD solver. The Clumon server provides information about job running on Teragra at NCSA analyzing scripts of PBS to e-AIRS Server and Web Portal Server. The eAIRS server contains e-AIRS middleware to monitor files produced by CFD solver on Teragrid Resources and automatically transfer result files from Teragrid resources to the mass storage at NCSA as known MassFTP [11]. Teragrid Resources are clusters which CFD solver actually runs on and The mass storage is massive storage at NCSA which MassFTP is running on.

More Detailed information on each part will be described in following sections.

IV. E-AIRS MIDDLEWARE

e-AIRS Middleware mainly consists of database, file transfer, monitoring, network message processing part and daemon. In the database part, metadata of result, residual and time survey files and job information are stored. In the file transfer part, file upload, download and 3rd party transfer are performed. The monitoring part observes result, residual and time survey files. Also the network message processing part is for interlocking web portals. Finally, the daemon manages job submitted by portal and executes this components by each job.

A. Database

The database component has functions that creates DB connection and execute insert, update and delete command of information to Postgresql DB by using JDBC. Information, such as a stored server name, a path, a size, owner, etc, about result files which is transferred to the mass storage after being produced by CFD solver and job information from web portals are sent is recorded in the database

B. File Monitoring

After the CFD solver performs computations by the period of regular iterations, result, residual and time survey files are created. These result files size are from several tens MB to several hundreds MB. Therefore sometimes the user's quarter can be exceeded by these result files. In order to solve quarter overflows, the file monitoring component observes creation. As we know the status of computation, this also observes update. If files are created or updated, files are transferred to the dedicated storage.

1) Result File Monitoring

The result file monitoring component checks the working directory of the working host and supervises creation of result files which registered in a configuration file. The e-AIRS middleware is not located in Teragrid resources. Therefore, the working directory's files are checked as the monitoring component gets the file list of working directory via GridFTP by periods. If registered results files are created, the files are transferred to the dedicated directory of the dedicated storage by 3rd party transfer via GridFTP. At this time, newly created directory's name in the storage is the job name which is included job information received by portal. After transferring files, this stores metadata of files into DB and delete result files in Teragrid resources. By using this result file monitoring, we can prohibit an overflow of user's quarter and manage the result files by each jobs.

2) Residual and Time Survey Files Monitoring

After computations are performed, not only result files, but also residual and time survey files and information file about computation are created. As progressing computations, these files are added to the existing files. Also these are transferred by the mass storage then information of these is updated to the DB.

C. File Transfer

The file transfer component basically provides upload, download in original e-AIRS middleware. 3rd party transfer to transfer result and residual and time survey files are added because the Teragrid resources where CFD solver runs are different from the server where e-AIRS middleware runs. When the file monitoring component sends the file transfer component a location of source files and destination files, this initializes GridFTP server of source and destination then start file transfer. At this time, a certification of the source and destination server must be same.

D. Network & Daemon

The network & daemon component provide TCP/IP socket communication to interlock web portals. A network client processes messages then retrieves job information or file transfer information. Daemon is waiting for connection and messages from network client. If network messages from web portals are about job's start, this creates job instance and put into Job Lists. This manages Job List and executes a file monitoring module by using job information of an infinite loop thread which is started with an e-AIRS system concurrently.

V. COLLABORATION BETWEEN JOB SUBMISSION OF GRID PORTLETS AND JOB MONITORING WITH CLUMON

In this system Job submission portlet uses Gram of Globus to submit job. After Gram submits job to PBS the job is actually running on Teragrid resources. In this mechanism between Gran and PBS Gram receives information that job is suspending, running, fail to submit. However Gram is not given more precise information about job. Though job is finished or goes down Gram informs a user that Job is done. Accordingly it is necessary that Gram is collaborated with PBS for monitoring job actually running on Teragrid. In this paper we implemented job monitoring mechanism as in Fig. 5.

The job submission portlet sends PBS Gram RSL which contains unique job id generated by our implementing in environment field to be identified by Clumon when job is submitted by Gram. After Clumon working at NCSA analyzes PBS RSL Script which is generated by Gram RSL, then detect unique jod ID in environment filed of RSL. This unique job ID is stored in Clumon's database corresponding with PBS Job ID.

On the other hand, the unique job generated by our implementation in job submission portlet is consisted of network message with other information such as Gram Job ID, user ID, grid credential and the path of CFD solver, so on. Then this network message is sent to e-AIRS middleware and stored as job information in database.

After this processing the user accesses their job information in the job monitoring portlet. When a user selects their job in the portlet, this connects to e-AIRS database and retrieves all job information with the user ID. The job information searched has also unique job ID, so this asks Clumon job information via HTTP Get method with uniqued job ID. When Clumon is received the request to retrieve job information Clumon searches PBS job information corresponding with unique job ID from Clumon's database, then this job information is returned to the portlet via HTTP Response method. This job information is displayed to the user in the portlet.

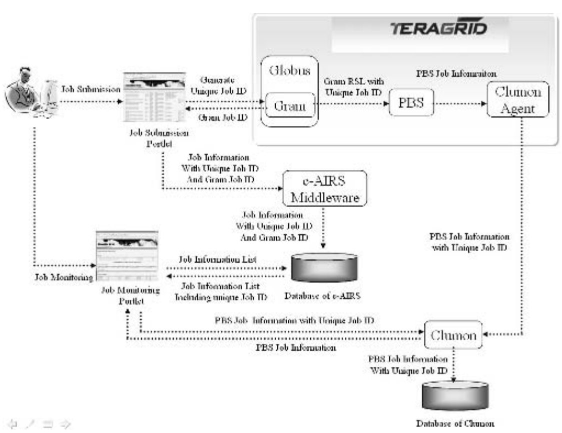


Fig. 5 Flow Diagram of Job Monitoring using Clumon

VI. CUSTOMIZED PORTLETS FOR CFD SOLVER

Two additional portlets, data management portlet and Graph portlet, which are not included in the existing Grid Portlets are developed for the e-AIRS and its CFD solver. The Data management ortlet shows the result files generated by the CFD solver and receives a file transfer request. And using the graph portlet users can monitor the status of computation through the several graphs that are based on the residual and time survey files.

A. Data Management Portlet

Information about the result file according to user's job is accessed through the portal. When a user selects the data management portlet, data management portlet fetches the job list and brief information on jobs from the database and displays it. When one job is selected from the list, web server downloads the metadata of result files corresponding to the job from the database and lists the file names. And then the user can request the portlet to download files by selecting the necessary files and specifying the download directory. After requesting, the portal builds a message using the user information and user's host address and sends the message to the e-AIRS middleware using the Network Client. e-AIRS middleware analyzes the message and transfers the requested files from the mass storage to user's computer by GridFTP 3rd party transfer. Following picture represents the process of data management & file transfer. This flow is showed in Fig. 6.

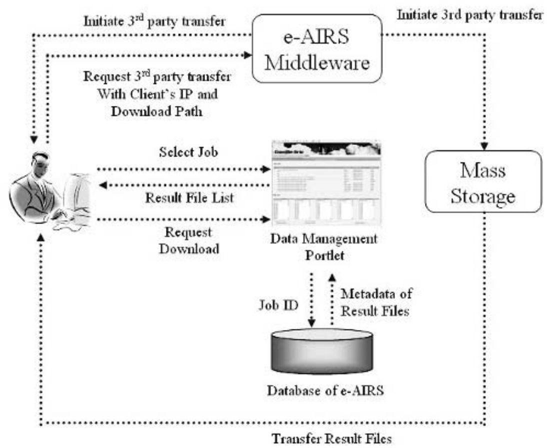


Fig. 6 Flow Diagram of file transfer from mass storage to client PC using 3rd party transfer

B. Graph Portlet

The graph portlet displays graphs that represent the result computed until the specified time, for users to monitor the status of their computations as in Fig. 7. When the user selects a job in job list, web server fetches metadata of residual and time survey files from the database corresponding with the selected job and downloads these files via GridFTP. The user can examine residual graph and time survey graph concurrently. The first graph represents residual graph. And the other is time survey graph which represents total run time, time step, total computation time, total communication time by the user's selection. With this graph service, the user can easily know the status of computation progress.

This page intentionally blank

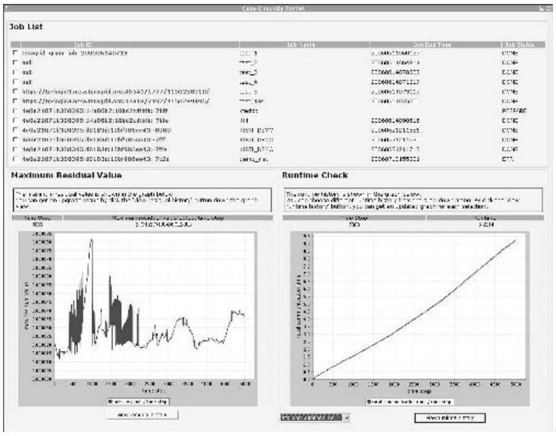


Fig. 7 Graph Portlet

VII. SUMMARY AND FUTURE WORK

In this paper, we adapted the e-AIRS system to the Teragrid at NCSA. We used and customized Grid Portlets which had been existed to access grid resources and submit job. And keeping track of a job connected with Clumon as a job monitoring tool a user can monitor job via portal. Also a user can maintain storage capacity of user and download result data from storage to one's computer via 3rd party transfer. Finally a user can check the status of computation progress easily by graphs in Graph service. In conclusion these components developed in this project offer aerodynamic researchers more easy-to-use research environment.

Hereafter we improve the speed of 3rd party transfer between servers. And we will adapt the e-AIRS Mesh which is added the function of transfer the CAD into the Mesh, the e-AIRS Viewer as a visualization tool and the parametric study service developing on the e-AIRS now to the Teragrid. And then, we will provide the total e-AIRS system on the Teragrid for aerodynamic research.

REFERENCES

[1] UK e-Science, http://www.nesc.ac.uk
[2] Cyberinfrastructure, http://www.nsf.gov/dir/index.jsp?org=OCI
[3] e-AIRS, http://obiwan.kisti.re.kr/escience/eairs
[4] Sang Beom Lim, Joobum Kim, Nam Gyu Kim, June H. Lee, Chongam Kim and Yoonhee Kim, "A Grid Web Portal for Aerospace", GCA'06, 2006, pp. 191-196.
[5] Yoonhee Kim, Eun-kyung Kim, Jeu Young Kim, Jung-hyun Cho, Chongam Kim, and Kum Won Cho, "e-AIRS: An e-Science Collaboration Portal for Aerospace Applications", HPCC 2006, LNCS 4208, 2006, pp. 813 – 822.
[6] Teragrid, http://www.teragrid.org
[7] Clumon, http://clumon.ncsa.uiuc.edu/
[8] J. Novotny, M. Russell, and O. Wehrens, "GridSphere: a portal framework for building collaborations", Concurrency: Practice and Experience, 2004, 5(16): pp. 503-513, http://www.gridsphere.org/gridsphere/gridsphere
[9] Globus, www.globus.org
[10] Apache Tomcat, http://tomcat.apache.org
[11] MassFTP, http://www.ncsa.uiuc.edu/UserInfo/Data/UniTree

Role of Microcracking in Cortical Bones

Natalia B. Romalis
Department of Mathematics
University of Bridgeport, Bridgeport, CT 06601

Summary

The influence of the microdefect interactions on the stress intensity factors (SIF's) at the tip of the macrocracks in cortical bones is considered using one of the linear fracture mechanics models developed by Tamuzs *at el* [1].

A human bone is a living organism governed by quite complex biological processes. However, at any given moment of time a frame of the human body is a mechanical structure of composite materials (of various mechanical properties for the different types of bones). As it is established experimentally [2], microcracks and other microdefects are present in cortical bones at every stage of life of the bone. They play a vital role in the process of remodeling of the bone.

With age, the distances between microcracks are decreasing [10], making "bridges" (uncracked ligament) smaller that causes increase of SIF's and decrease of the critical fracture load. In another words , a bone fracture can occur even without extraordinary external force.

The objective of this work is to provide a mechanistic interpretation of interaction of macro- and microcracks in cortical bones. The structural integrity of a composite material of the bone is obviously vital for the work of the human body frame. Traditionally, the bone quality is defined in terms of bone mineral density (BMD), namely, the amount of bone mineral per unit of volume. The fracture of the bone may occur not only as a result of a single impact but may also be caused by the microcrack interactions resulting in so called stress fracture (without a major impact). Therefore, for comprehensive understanding of the bone fracture it is necessary to study the microstructural interactions in bones.

As the linear fracture mechanics is considered by most researchers an appropriate tool for investigating failure processes in bones [2], the analysis of the stress intensity factors governing the hierarchy in macro-microcrack propagation and the amplification ("intrinsic") and shielding ("extrinsic") effects is conducted using solutions obtained in [1,4].

Introduction

The fracture of bones has been attracting the interest of scientists for many years because of the health concerns and because of the fact that fracture mechanics can explain basic mechanisms of the bone failure. During the last two decades the number of models of macro-microdamage interactions was developed for various types of non-homogeneous materials.

The competing mechanisms of the shielding and amplification roles of the microdamage in artificial non-homogeneous materials were discussed in the review article [3]. However, the fracture of the bones differs significantly from the fracture of other composite materials because of the self-repairing ability of bones through the mechanism of the damage-stimulated remodeling as described in the comprehensive review [2].

Statement of the Problem

During the life-time of the bone, the competing mechanisms of the bone resorption and formation change the microstructure of the bone. A significant amount of research (reviewed in [2]) was conducted that demonstrated that a microdamage may, in fact,

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 359–361.

increase the crack resistance in a cortical bone. The linear elastic fracture mechanics approaches for solids with microdefects [1,4] are used to analyze the intrinsic and extrinsic effects of the macro-microcracks interactions.

Although during the everyday activities bones are subject to 3-D loadings that may result in the modes I, II, or III fractures, “cortical bone shows the least resistance to fracture under mode I (purely tensile) loading, and, accordingly, this loading mode has received the most attention in the literature”, as stated in [2]. It is well established in the fracture mechanics of microdamaged materials that in the presence of microcracks, the crack extension is governed by the mutual competition of the amplification and shielding roles of the arrays of microcracks and their influence on a macrocrack (see, for example [1, 4, 5, 6]).

There is a number of relatively recent publications on extrinsic toughening mechanism behind the crack tip that attributes it primarily to the crack bridging when unbroken parts of the bone partially sustain the applied load (see, for example, [7, 8]).There are different approaches for accounting for the contribution of the crack bridging in bones to the stress intensity factor (SIF) (see, for example, [9,10]).

In present work the modeling problem for the SIF in the presence of the crack bridging was solved using asymptotic solution obtained in [4] for the interaction of the main crack and arbitrary distribution of discrete microcracks.

Solution to the Problem

The SIF at the tip of the main crack with microcracks separated by the bridging of undamaged material (schematically represented in Fig.1) can be evaluated by the formula

$$K = \left\{ 1 + \frac{\lambda^2}{4} \sum_{k=1}^{N} \left[J_{k0} \left(\mathrm{Re} \frac{2}{(w_k - 1)\sqrt{w_k^2 - 1}} \right) + \overline{J_{k0}} \frac{(\overline{w}_k - w_k)(2w_k + 1)}{\overline{(w_k - 1)}(\overline{w}_k^2 - 1)^{3/2}} \right] \right\}; \quad (1)$$

where $J_{k0} = \frac{1}{2}\left[\mathrm{Re}\left(\frac{2w_k}{\sqrt{w_k^2 - 1}} \right) + \frac{(\overline{w}_k - w_k)}{(\overline{w}_k^2 - 1)^{3/2}} \right]$;

$K = \frac{k_I - ik_{II}}{k_{cr}}$ is a dimensionless SIF representing the ratio of the SIF at the crack tip to the critical SIF providing a start to the main crack growth,

λ is the ratio of the length of the microcrack to the length of the main crack,

w_k is the complex coordinate of the center of the microcrack in the global coordinate system with the origin at the center of the microcrack that can be easily represented in terms of the distances d_k between adjacent microcracks.

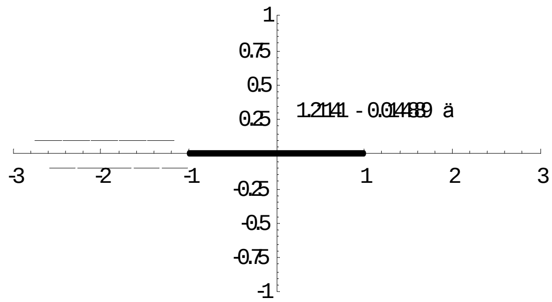


Fig.1 Mutual macro-microcrack orientation

The mutual orientation of the macrocrack and microcracks separated by the bridging is depicted on Fig.1. Calculations were performed on *Mathematica* for various distances between microcracks. Graphs of dependence of the dimensionless SIF’s on distances between microcracks for $\lambda = 0.2$ and $\lambda = 0.3$ are represented on Figs.2,3

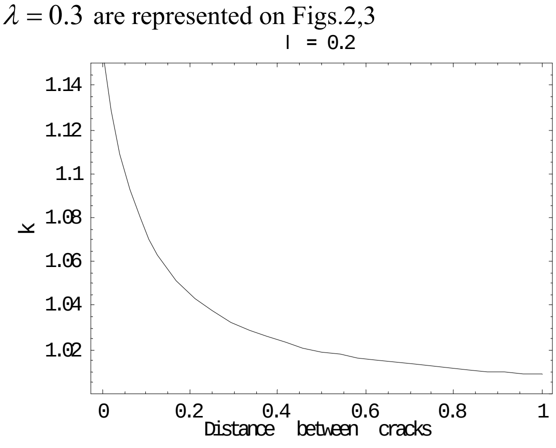


Fig.2 Dimensionless SIF as function of dimensionless size of the microcrack

This page intentionally blank

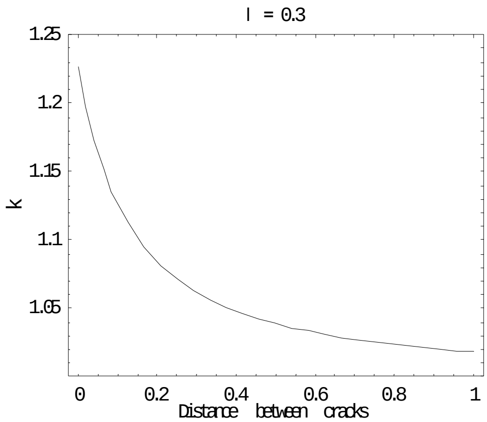


Fig.3 Dimensionless SIF as function of dimensionless size of the microcrack

Discussion

As can be seen from graphs in Figs. 2,3, the size of the bridging (distances between adjacent microcracks) determines the SIF that in its turn determines a macrocrack initiation toughness. Experimental data presented in [2] (based on its authors research and on data from [6] and [9]) show a significant decrease of the crack initiation toughness with age for human cortical bone. One of the possible reasons for this can be a smaller size of the uncracked ligament (distance between microcracks) for aged bones observed in experiments [10].

Conclusion

One of the mathematical models for evaluation of SIF's at the macrocrack's tip in the array of the microcracks presented in [1] was applied for analysis of shielding effect of microcracks in bones. Present work will be continued to obtain explicit relations between SIF's (and critical loads of the macrocrack initiation) and the age of the bones.

Although biological and morphological processes of the bone formation and remodeling are very complex, at any given moment of time the frame of a human body is a mechanical structure of composite materials. Thus, the fracture criteria and a set of critical loads can be found. Such explicit formulas for bounds of critical loads would be quite useful for determining such things as proper level of exercise and physical activities for various groups of population.

Reference

[1] Tamuzs,V., N.Romalis, and V.Petrova (2001), Fracture of Solids with Microdefects, *Nova Science Publishers*, 238p.
[2] Ritchie, R.O., J.N. Kinney, J.J.Kruzic, and R.K.Nalla (2004), A Fracture Mechanics and Mechanistic Approach to the Failure of Cortical Bone, *Fatigue Fracture Engineering Material Structure*, 28, 345-371
[3] Petrova, V., V.Tamuzs, and N.Romalis (2000), A Survey of Macro-Microcrack Interaction Problems, *Appl. Mech.Rev*., Vol 53, No.5, pp. 117-145
[4] Romalis,N.and V.Tamuzs (1984), Propagation of a Main Crack in a Body with Distributed Microcracks (Russian) *Mekh. Komposit Mater.,* No.1, pp. 42-51 (Engl.Transl: *Mech. Composite Materials*, pp. 35-43).
[5] Ritchie, R.O. (1988), Mechanism of fatigue crack propagation in metals, ceramics and composites: Role of crack-tip shielding. *Mater.Sci.Engng* 103, 15-28.
[6] Vasishth, D., Behiri, J.C. and Bonefield, W. (1997), Crack growth resistance in cortical bone: Concept of microcrack toughening. *J.Biomech*., 30, 763-769
[7] Norman, T.L., Vashishth, D.,and Burr, D.B. (1995) Fracture toughness of human bone under tension. *J.Biomech*., 28, 309-320
[8] Nalla, R.K., Kruzik, J.J., Kinney, J.H. and Ritchie, R.O.(2005) Mechanistics aspects of fracture and R-curve behaviour in human cortical bone, *Biomater*. 26, 217-231
[9] Wu,P.C. and Vashishth, D. (2002) Age related changes in cortical bone toughness. In: *2nd Joint EMBS/BMES Conference, IEEE*, Houston, Texas, USA, 425-426
[10] Nalla, R.K., Kruzik, J.J., Kinney,J.H. and Ritchie, R.O.(2004) Effect of aging on the toughness of human cortical bone: Evaluations by R-curves, *Bone*, 35, , 1240-1246

Algorithm for Solving the Collisions in Uniform Simple Hashing with Combined Linked Ordered Lists

Tudor Nicoleta Liviana
Department of Computer Science
Petroleum-Gas University of Ploiesti,
39 Bucuresti Avenue, 100680, Ploiesti, Romania
tudorl@upg-ploiesti.ro

***Abstract*- This article addresses the problem of solving the collisions through chaining in the simple uniform hashing. The personal contribution consists of the improvement of the algorithm of dispersion through linking, by modifying the linking schema of the elements of the hash table. A contribution presents solving the collisions by creating combined linked ordered lists for storing the elements that have the same hash value. The code returned by the second hash function applied to the key of each element, is used for ordering the elements of the combined linked list. The proof of the performance of the proposed algorithm consists in the computing of the time needed for the search with and without success and of the comparative analysis of the time of execution of the algorithms.**

Keywords: algorithm dispersion, hash table, collision, combined linked ordered lists, simple uniform hashing, linking.

I. Introduction

This article presents the way of solving the collisions through linkage, in uniform, simple dispersion and presents several improvements. First of all, we are introducing the notion of combined linked ordered lists (CLOL) and we are presenting the algorithm of searching in a hash table that uses CLOL. For comparing two elements, we propose the usage of the code generated by the second hash function. Second, we are proving the fact that CLOL offers a better search time than the linked lists. Third, we are analyzing the statistics of the execution times for the presented algorithms that are dependent of the load factor of the hash table.

The paper is organized as follows:

- the section The Hashing with Chaining addresses the issue of the hashing of keys from a domain of possible keys to pseudo-random locations and the way of avoiding the collisions through linkage in the case of simple, uniform hashing;
- the section Solving the Collisions using Combined Linked Ordered Lists presents a way of optimizing the hashing with chaining algorithm by modifying the linking schema of the elements of the hash table. The idea is to create combined linked ordered lists for storing the elements that have the same hash value. The proof of the efficiency of the CLOL algorithm is based on computing the time needed for a search, with and without success (when the value exists and when it does not exist in the collection).
- the section Application: Processes Monitoring on the Computer describes an application of the Hash tables using CLOL.
- the section The Comparative Analysis of the Execution of Algorithms is dedicated to interpretation of the statistics regarding the execution time of the algorithms presented in the previous sections.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 363–368.

II. THE HASHING WITH CHAINING

A hash table [1] is an efficient data structure that can store numerical data, strings or complex objects in pseudo-random locations, usually by using static structures of type array, or dynamic structures as linked lists. Each element of the array is called slot (or bucket) and can contain a pair key-value or a record.

For defining the hashing [2], we consider D the domain of the possible keys and E the set of the keys that are actually stored. The hashing means that an element having the key k is stored at location h(k), where h is a function of hashing defined as follows [3]:

$h: D \rightarrow \{ 0, 1, \ldots, n-1\}, |T| = n$
$k \rightarrow h[k]$, where h[k] is an index in the hash table $T[0 \ldots n-1]$.

If there are two keys for which the same index is allocated, than the corresponding records can not be stored at the same location. There are lots of techniques for avoiding the collision (the repetition of indexes) and among those, the most popular are the linkage and the direct addressing.

Next, we are going to present the way of avoiding the collisions through linkage in the case of simple, uniform hashing.

Each slot of the hash table points to a linked list of inserted records that point to the same element (figure 1) or is NULL, if it does not contain any elements [4].

The base operations supported by the hash tables, in the case of linkage, are:

- Inserting – each record (or pair key – value) can be inserted in a free slot of the table or in a slot that points to a linked list of records;
- Searching (key) that returns true/false if the key is found/not found in the table;
- Deleting (key).

The algorithm that creates and displays a hash table using the linkage, contains the following procedures and functions for inserting, searching and deleting:

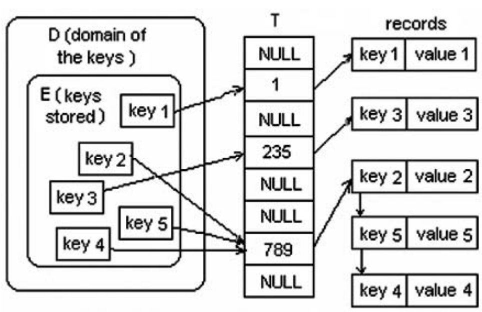


Fig. 1. Hash table with linked lists

```
procedure inserting ( x, k )
    // k = h(key[x])
    Insert x in Hash table at position k, at the
    begining of the list
end // procedure
```

```
function exist(k, c)
    if Hash table is empty (at position k)
            return(false)
        else
            q ← element k of the Hash table
            while (q <> NULL)
                    if key[q] = c
                            return (true)
                    endif
                    q ← next[q]
            repeat
            return(false)
    endif
    end // function
```

```
procedure delete( x)
    Delete x from list, from position h(key[x])
end
```

For the analysis of the time needed for the hashing of the elements in hash tables, one needs to find the load factor of the table which will be defined as follows:

Definition 1: Given a hash table T with n elements (T[n]), that stores m elements, the load factor α for the table T is defined as follows: $\alpha = m / n$ and represents the average number of elements stored in a linkage.

In what follows, we are presenting several base definitions of the used notions [5].

Definition 2: *The simple, uniform hashing* defines the hypothesis according to which each element can be hashed in any of the n slots of the table with the same probability regardless of the slots where the other elements have been hashed.

Definition 3: (Asymptotic notation θ) Given a function g(n), we mark with $\theta(g(n))$, the set of functions [6]:

$$\theta(g(n)) = \{ f / (\exists)\, c_1, c_2, n_0 > 0, \text{ so that } 0 <= c_1 g(n) <= f(n) <= c_2\, g(n),\ (\forall)\, n >= n_0\}. \quad (1)$$

In order to find the average number of elements examined by the search algorithm, we are considering two cases: when the search is unsuccessful (no element from the table has the searched key) and the case when the search is successful, so that it finds an element with the given key. Cormen [7] has proven the following theorem:

Theorem 1: In a hash table where the collisions are solved through linkage, a successful search needs, in average, a time of $\theta(1+\alpha)$, in the hypothesis of a simple, uniform hashing.

The hashing through linkage can be optimized by changing the linkage schema of the elements from the hash table and by ordering the elements of the linked list, case which is described in the following section.

III. Algorithm for Solving the Collisions Using Combined Linked Ordered Lists

A personal contribution consists of the improvement of the linking algorithm (presented in the section above) through the development of an algorithm that creates a hash table with combined linked ordered lists for storing the elements that have the same hash value [8]. The notion of combined linked list will be introduced as follows:

Definition 4: (the combined linked list) Given X, the set of the information contained in the nodes of a list, with |X| =n and Y the set of pointers in any given space of addressable memory. A *combined linked list* is a set Lp = { ($pred_i$, $info_i$, urm_i) / $info_i \in X$, $pred_i$, $urm_i \in Y$, i = 1, …, n}, where over Lp there has been defined a direct and an inverse linear structure, $pred_1 = urm_n =$ NULL, $pred_i = info_{i-1}$, $\forall$ i = 2, …, n and $urm_i = info_{i+j}$, $\forall$ i > = 1 and 0 <= j <=2.

An example of combined linked list with 6 elements is presented in figure 2.

Notice: For the ordering of the elements in the combined linked list, the code returned by the second hash function applied to the key of each element is used.

The algorithm that creates a hash table with combined linked ordered lists (CLOL) for the storing of the elements that have the hashing values contains the following searching function:

```
function exist(k, c)
 if slot[k] = NULL
       return(false)
else
q ← slot[k]
if code[k] > code[c]
     // the code returned by the second hash function
return(false)
endif
// going through next link
for (q = slot[k]; q <> NULL; q ← next[q])
p ← q
if code[q] > code[c]
        goto et1
endif
if key[q] = key[c]
                  return (true)
endif
repeat
et1:  // going through previous link
for( ; p!= NULL; p  ← previous[p])
if code[p] < code[c]
         goto et2
endif
if key[p] = key[c]
return(true)
endif
repeat
et2:  return(false)
endif
end // function
```

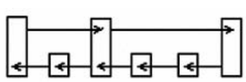


Fig. 2. Combined linked list

Sentence 1: Let there be a hash table T with |T| = n, that stores m elements and uses combined linked ordered lists for storing the elements that have the same hash values. In the table T, where the collisions are solved through linkage, a search requires a time equal to $\theta\left(1+\alpha\frac{m-1}{6m}\right)$, given the hypothesis of a simple, uniform hashing, where $\alpha = m/n$ is the load factor of the table T.

Proof: It's considered that each element can be hashed in any of the n slots of the table T with the same probability.

For a key k from the domain of the possible keys, the value h(k) can be computed in a time $\theta(1)$. The time needed for searching an element with the key k, is linear dependent on the number of elements of the ordered linked list T[h(k)].

Case 1: successful search - We examine the average of the m objects of the table as being 1+ the average length of the combined linked ordered list, to which we add the i^{th} element, that is at most equal to $\frac{i-1}{3n}$ (figure 3):

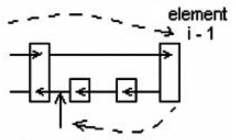


Fig. 3. Searching

It results that the average number of compared elements is:

$$\frac{1}{m}\sum_{i=1}^{m}\left(1+\frac{i-1}{3n}\right)=\frac{1}{m}\left(m+\frac{1}{3n}\sum_{i=1}^{m}(i-1)\right)=$$

$$1+\frac{1}{3mn}*\frac{(m-1)m}{2}=1+\alpha*\frac{(m-1)}{6m}. \qquad (2)$$

So, the total time needed for a successful search (including the time needed for computing the hash function) is:

$$\theta\left(2+\alpha\frac{m-1}{6m}\right)=\theta\left(1+\alpha\frac{m-1}{6m}\right). \qquad (3)$$

Case 2: unsuccessful search

In an unsuccessful search, the algorithm passes through the whole combined linked ordered list by following the next link. The same way, the average length of the combined linked ordered list to which we try to add the element i is $\frac{i-1}{3n}$, and the average number of compared elements is:: $1+\alpha*\frac{(m-1)}{6m}$. So, the total time needed for an unsuccessful search (including the time needed for computing the hash function) is:

$$\theta\left(2+\alpha\frac{m-1}{6m}\right)=\theta\left(1+\alpha\frac{m-1}{6m}\right). \qquad (4)$$

Corollary 1: Let there be a hash table T with |T| = n, that stores m elements and uses combined linked ordered lists for storing the elements that have the same hash values. In the table T, in which the collisions are solved through linkage, the total time needed for a search is smaller than $\theta(1+\alpha)$, in the hypothesis of a simple, uniform dispersion, where $\alpha=\frac{\mathrm{m}}{\mathrm{n}}$ is the load factor of the table T.

Proof: The total time needed for a successful or unsuccessful search is $\theta\left(1+\alpha\frac{m-1}{6m}\right)<\theta(1+\alpha)$, where m>0 and $\alpha=\frac{\mathrm{m}}{\mathrm{n}}$.

IV. Application: Processes Monitoring on the Computer

This section presents an application for processes monitoring on the machine, that implements objects cache managment pattern. The cache is implemented by a Hash table and the Hash table values are CLOL (combined linked ordered lists) objects that refer to the process object.

Caching enables applications to avoid issuing multiple read operations for the same data item. Every client program that retrieves objects utilizes the object cache [9]. A client-side object cache is allocated for every environment handle initialized in object mode. Multiple threads of a process share the same client-side cache [10, 11] using the same environment handle. One copy of each referenceable object exists in the cache for each connection.

Objects that are no longer needed can be freed; they can be swapped out of the cache [12], freeing the memory space occupied. Deciding which and how many objects to keep in memory is called cache management [13]. Description of the cache management pattern is the following:

- Client - instances of classes that obtains access of specified objects to cache manager
- Cache manager - client requests objects
- Cache - a cache object manages a collection of cached objects. Objects cache is a Hash table, with elements CLOL (combined linked ordered lists)
- Object - identifies the object to be fetched or created.

The figure 4 shows the functioning of cache:

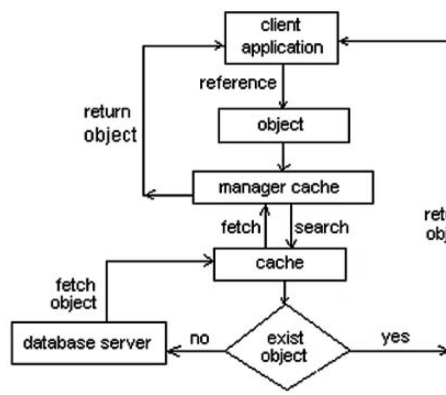


Fig. 4. Issuance of objects cache

The pseudo code for the cache management algorithm is the following:

```
procedure cache management
search object within cache
if object is not found within cache
 search object on the disk (on the database server)
 if object is found on the disk
if cache is full
remove the last element
endif
add object to cache on the first place
endif
   else
   object becomes first in cache
 endif
 return object
```

The *CacheManager()* procedure performes the following:

- fetch an process object for the ProcessId from the cache or server if it is not in the cache
- return process object or null if process object not found
- the cache uses a combined linked ordered lists to determine the most recently used process object.
- the cache is implemented by a Hash table. The Hash table values are combined linked ordered lists objects that refer to the actual process object.

V. COMPARATIVE ANALYSIS OF THE EXECUTION TIME OF THE ALGORITHMS

The search of an element in the hash table is determinant [14] for the execution time of the algorithm; the insertion and the deletion are executed in a constant time $\theta(1)$.

The table 1. presents the results of evaluating the search time for the two algorithms: the simple, uniform dispersion with linked lists and the simple, uniform dispersion with CLOL.

TABLE 1.
EVALUATING THE SEARCH TIME

n	M	Load factor $\alpha = \frac{m}{n}$	search time for linked lists (ms)	search time for CLOL (ms)
31	4	0.133	1.133	1.016
31	25	0.833	1.833	1.133
31	30	1	2	1.161
31	45	1.5	2.5	1.244
31	60	2	3	1.327

- data structure determine which process object will be removed from the cache when a process object will be added to a full cache.

An evaluation of the effectiveness of caching can be obtained computing a statistic hit rate. The hit rate (H) is the percentage of object fetch requests that the cache manager satisfies [15] and it depends on the implementation of the cache management pattern. The miss ratio (M) is calculated as:

$$M = 1 - H \tag{5}$$

The search time in cache (implemented by a Hash table with CLOL) determines the read time from cache. If hit time is Th (read time from cache), and miss time is Tm, then the average time to access memory cache can be calculated as:

$$T = Th * H + Tm * M \tag{6}$$

The Th/Tm ratio influences effectiveness of caching and Tm can be expressed as: Tm = Th * k, k >= 0.

Then the average time T is:

$$T = Th * H + Th * k * (1 - H) = Th (H + k * (1 - H)), k >= 0. \tag{7}$$

The efficiency of the proposed algorithm can be depicted through the statistics regarding the execution time or the search time of an element in the case of the algorithm with linked lists compared to the algorithm that uses CLOL, as stated in the next paragraph.

After having compared the values of the two last columns of the table, we can see the load factor of the hash table being the same, the searching time in a combined linked ordered list being less than the searching time in linked lists. The same way, we can observe the evolution of the searching time from a load of 25% of the table up to 200% (figure 5).

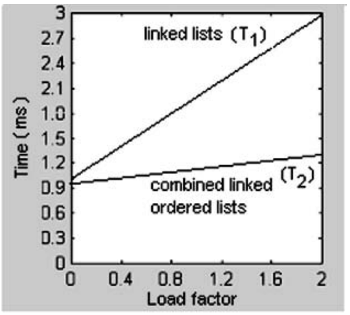


Fig. 5. Search time

There is a linear dependency of the time, on the load factor, but the slope of the line differs, from 1 (it forms a 45 degrees angle with the X axis for linked lists) up to 1/6 – 1/(6m). If n $\rightarrow \infty$ (if it is big enough) and $\alpha \rightarrow 1$, than the line's slope -> 1/6 (an angle of 15 degrees with the X axis for CLOL).

We can also conclude that a search in the hash tables has a higher performance using CLOL than using linked lists.

VI. Conclusion

In uniform, simple hashing the algorithm of linkage with CLOL requests less searching time than in the case of linked lists. The efficiency of the searching function is determined by the ordering of the elements in the combined linked list using a second hash function applied to the key of each element, but by the changing of the linking schema for the elements as well. For proving the performance of the proposed algorithm, we have computed the time needed for a successful and an unsuccessful search and we have compared the execution times of the algorithms. The efficiency of the algorithm with linking with CLOL was highlighted in an application of monitoring a computer's processes, which creates a cache of process objects using hash tables.

References

[1] G. H. Gonnet**, „**Handbook of Algorithms and Data Structures", *Addison-Wesley*: pp. 200-286, 1984

[2] A. Apostolico, F. P. Preparata, „Data Structures and Algorithms for the String Statistics Problem", *Algorithmica*: pp. 481-494, 1996

[3] M. Dietzfelbinger, A. Karlin, K. Mehlhorn, F. M. Heide, H. Rohnert, R. E. Tarjan, „Dynamic perfect hashing: Upper and lower bounds", *SIAM Journal on Computing 23*, 4, pp. 738–761, 1994

[4] D. Knuth, *„The Art of Computer Programming"*, vol. 3, *Addison-Wesley*, 1973

[5] D. E. Knuth, „*The art of computer programming*", Vol. 2, Seminumerical algorithms. Third ed. Reading, Mass.: Addison-Wesley, 1998

[6] M. J. Franklin, M. J. Carey, M. Livny, „Transactional client-server cache consistency: Alternatives and performance", *ACM Transactions on Database Systems 22*, 1997

[7] T. Cormen, C. Leiserson, R. Rivest, „Introducere in algoritmi", *Agora*, pp. 192, 1990

[8] M. Mor, S. Fraenkel, „A Hash Code Method for Detecting and Correcting Spelling Errors", *Comm. of the ACM*, pp. 935 938, 1982

[9] http://www.squid-cache.org

[10] M. D. Hill, *Aspects of Cache Memory and Instruction Buffer Performance,* PhD thesis, Unversity of Berkeley, 1987

[11] Z. Chen, Y. Zhou, K. Li, „Eviction-based cache placement for storage caches", In *Proceedings of USENIX Technical Conference*, San Antonio, 2003

[12] A. Shatdal, C. Kant, J. F. Naughton, „Cache Conscious Algorithms for Relational Query Processing", In *Proceedings of the 20th VLDB*, pp 510-521, 1994

[13] M. Grant, „Pattern Summaries: Cache Management*",* *Jupitermedia Corporation*, 2006

[14] G. N. Frederickson, „Searching Intervals and Compact Routing Tables", *Algorithmica,* pp. 448-466, 1996

[15] T. Wong, J. Wilkes, „My cache or yours? making storage more exclusive*",* In *Proceedings of the Usenix Technical Conference*, 2002

A Comparative Study Regarding a Memory Hierarchy with the CDLR SPEC 2000 Simulator

O. Novac [*], M. Vlăduțiu [**] , St. Vari Kakas [*], Mihaela Novac[***] and M. Gordan [***]
[*] Department of Computers, University of Oradea,
[**] Department of Computers, "Politehnica" University of Timişoara,
[***] Department of Electrotechnics, University of Oradea

***Abstract*-We have built a simulator named, CDLR SPEC 2000. This simulator is based on traces for systems with cache memories. With CDLR SPEC 2000 program we can study, for a memory hierarcy, the next parameters: maping function, block size, writing strategies, replacement algoritm, size of cache memory, number of cache sets (for the set associative caches), number of words form a block. The simulator can be used for the study of cache memory behaviour. Also the CDLR SPEC 2000 program, introduce the calculus of CDLR of a memory hierarchy.**

A common metric used for the assessment of overall reliability, in a memory hierarchy is the Mean Time To Failure (MTTF), but it doesn't take into account for the time of data storage in each level. We propose another metric, Cache Data Loss Rate (CDLR), for this reason. We will derive a recurrent formula for computing CDLR, and we will validate it by computer simulation.

I. INTRODUCTION

All high performance microprocessors use a hierarchy of cache memories to hide the slow access to the main memory [1]. Hence the memory hierarchy is important part in a computer system With each new generation of integrated circuit technology, feature sizes shrink, creating room for more functionality on one chip. We see a fast growth in the amount of cache that designers integrate into a microprocessor to gain higher performance. Hence, caches occupy a substantial area of a microprocessor chip and contain a high percentage of the total number of transistors in the chip. Consequently, the reliability of the cache has a big impact on the overall chip reliability [5]. Practically if a computer system has a fast and efficient processor, if the memory hierarchy is slow and inefficient, memory access bottlenecks will arise and the overall system performance will be low. For this reason in any system design an adequate attention should be paid to the design of memory hierarchy subsystems, such as memory interface, cache, paging mechanism, TLB, CPU memory hierarchy support registers.

Caches are used to bridge the speed gap between microprocessors and main memory. Without them, the processor can still operate correctly but at a substantially lower performance. Most of the architectures are designed such that the processor can operate without a cache under certain circumstances. Therefore, the obvious solution to a faulty cache is to totally disable it. In set-associative caches, a less extreme solution is to disable one unit (way) of the cache [4].

The memory devices of a computer system are organized in a hierarchy in order to achieve a good rate between performance and cost per bit. Proceeding down in the memory hierarchy, its reliability improves owing to the storage technology in different levels. There is of course a tradeoff between high reliability and performance, which is influenced beside the construction, by the transfer policy used among levels. Transfer policy is important because it directly affects the reliability of the overall hierarchy. The first possibility is to write through to the most reliable, lowest level every time a data is transmitted from the CPU. This policy offers good reliability, but bad performance (high overhead for transferring data). The second possibility is to write back data to lower level only when needed (for instance based on the amount of data in a level). This method yields a more reduced reliability (as data stays longer in a less reliable level), but better performance (less overhead for transferring data). The third possibility is the delayed write, when data is written from level L to level L+1 after a time denoted $delay_L$. So $delay_L$ is the age of the data before it leaves level L and is written to level L+1. We can observe that $delay_{L,}$ monotonically increase with L.

II. MTTDL AND CDLR METRICS

MTTDL (Mean Time To Data Los) is actually the MTTF (Mean Time To Failure), the mean time to a hierarchy breakdown. If we suppose a high serial reliability of the system's block diagram and for each level we have an exponential distribution for a given MTTF, the MTTDL for a hierarchy with N levels is given as in [2]:

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 369–372.

$$MTTDL = \frac{1}{\sum_{L=1}^{N} \frac{1}{MTTF_L}} \quad (1)$$

In computing the MTTDL we can observe that it is limited by the reliability of the least reliable level. One can say that MTTDL is relevant only when the data loss is proportional. The MTTDL is general and it can not distinguish between the loss of a great amount of data (due to a disk failure) and a data loss due to a memory error.

For this reason we will propose a new measure - CDLR (Cache Data Loss Rate) which represents the data loss ratio during the memory hierarchy failure. The mean access time for a reliable architecture with N levels is calculated as the arithmetical mean of access times:

$$\sum_{L=1}^{N} hitRate_L x Acces_L \quad (2)$$

The DLR (Data Loss Rate), for a reliable memory hierarchy is the weighted sum of the data loss duration when a level fails and is given by the following formula (3) [2]:

$$DLR = \sum_{L=1}^{n} \frac{duration_L}{MTTF_L} \quad (3)$$

We can say that $duration_L$ implies a special case, that there is no lower level in which data can be transferred.

A first method to tackle this problem, is to suppose that the lowest level has an infinite MTTF and using a stocking technology that closes to this ideal (relatively to the reliability of the other levels).

Another approach is that of attributing to $duration_L$ the quantity of useful data that might be lost in case of a level failure. For example perhaps only the last year's data might be useful in the case the magnetic tape.

Both approaches lead to the following relation:

$$\sum_{L=1}^{N} \frac{f_L}{MTTF_L} \quad (4)$$

where f_L represent the amount of data lost in case level L fails. We can consider for example that f_L is a linear function that depends to $length_L$, such that in this formula newer data have a larger proportion than the old data.

The previous formula does not count the collateral malfunctions. If a malfunction affects several levels (such a malfunction is flooding), it must be modified in order to count a type a malfunction once. Thus a single malfunction would drop the MTTF on several levels and will affect the $MTTD_L$ and DLR several times. The way of correction the correlated malfunctions are to calculate the MTTDL and the DLR values, by adding the types of malfunctions rather than the levels of reliability:

$$MTTDL = \frac{1}{\sum_{altypeofmalfunctionsF} \frac{1}{MTTF_F}} \quad (5)$$

$$DLR = \sum_{altypeofmalfunctionsF} \frac{malfunction_F}{MTTF_F} \quad (6)$$

where $MTTF_F$ is the average time till the apparition of an F type malfunction and $malfunction_F$ is the duration of data loss, when malfunction F appear.

We shall use other formula than the one used in [2], to compute CDLR. For this purpose we shall use the following principle [3]:

No. losses in L = No. losses due to L + No. losses Due in (L+1) [No. losses in L induced from a loss in (L+1) – No. losses already produced in L from those induced by (L+1)

Starting from this principle, we obtain a recurring formula that has a better accuracy than the one used in [2]:

$$CDLR_L = \frac{delay_L}{MTTF_L} + CDLR_{L+1}\left(1 - \frac{delay_L}{MTTF_L}\right) \quad (7)$$

where $CDLR_L$, is a new term and represents the rate of data lost in level 1. (this formula does not allow us to take into consideration errors, data what appear in a data field already corrupted on a previous level). As I mentioned, the last level is a special case, because it has no lower level to transmit data. For this reason, we ought to take into consideration $MTTF_N$ is infinite, which means $CDLR_N = 0$ (otherwise the CDLR of the entire hierarchy would be of 100% and thus all the data would be lost in time). The evaluation of performance and reliability of some hierarchy of memory can be made now, by using the two sizes.

The CDLR SPEC 2000 program is built as the SMP Cache 2.0 program [6] for working with Traces, the difference is the fact that the CDLR SPEC 2000 program use traces of SPEC 2000 benchmark. The parameters of the CDLR SPEC 2000 program can be modified, by the user.

III. SIMULATION EXAMPLE

For working with the simulator, it is necessary to use data files with the "calls" and memory addresses demanded by the processors during the execution of a program: the named memory traces. The trace files will allow us to emulate the programs with our CDLR SPEC 2000 program.

In case of a simulation with the CDLR SPEC 2000 program we must specify the context of simulation and the parameters for the simulation. (mapping function, block size, writing strategy and the replacement algorithm), in a input file. We will specify the mapping function: direct, set-associative or fully-associative. The second parameter we must specify is the block size, the third parameter we must specify are the writing strategies: write back (1) or write through (2). The last parameter is the replacement policies: LRU (1), LFU (2), Random (3) or FIFO (4).

We must specify also for each level of the memory hierarcy the next parameters: memory size, MTTF, fetch time, read time, write time and miss penalty. After all the parameters where specified we will load the trace file, and this file is selected from the trace files of SPEC 2000 benchmark suite. In the table I and table II, we have presented the structure of two input files (spec1.in and spec2.in).

TABLE I

Input file	spec1.in		
mapping function			4
block size			16
writing strategy			Back
replacement algorithm			FIFO
Level	I	II	III
Memory size	4096	16384	32768
MTTF	1000000	9000000	90000000
fetch time	1 ns	10 ns	20 ns
read time	3 ns	20 ns	40 ns
write time	5 ns	50 ns	90 ns
miss penalty	10 ns	150 ns	300 ns
Trace			Swim_m2b

Our CDLR SPEC 2000 simulator, calculate the CDLR parameter, for the memory hierarchy. In table I we present a memory hierarchy with the following parameters: the mapping function is 4 way set-associative, the size of a block is 16, the writing strategy is write back (1) and the replacement policy is FIFO (4). The parameters for the first level of the memory hierarchy are: the size of memory is 4096 bytes, the MTTF is 1000000s, the fetch time is 1ns, the read time is 3ns, write time is 5ns and miss penalty is 10ns. The parameters for the second level of the memory hierarchy are: the size of memory is 16384 bytes, the MTTF is 90000000s, the fetch time is 10ns, the read time is 20ns, write time is 50ns and miss penalty is 150ns. The parameters for the third level of the memory hierarchy are: the size of memory is 32768 bytes, the MTTF is 900000000s, the fetch time is 20ns, the read time is 40ns, write time is 90ns and miss penalty is 300ns.

The structure of the first output file (spec1.out), obtained after the computation of the CDLR SPEC 2000 with the first input file spec1.in (presented in Table I), is presented bellow :

----------------[LEVEL 1]--------------------------
MTTF Level : 1000000
Delay Average Level: 33516
Times Level: 519963

----------------[LEVEL 2]--------------------------
MTTF Level : 90000000
Delay Average Level: 3034236
Times Level: 22642

----------------[LEVEL 3]--------------------------
MTTF Level : 900000000
Delay Average Level: 31605791
Times Level: 1680
MTTF hierarcy: 987925
CDLR hierarcy: 6.764735

The CDLR parameter and the global MTTF for the memory hierarchy presented in table I are:
CDLR = 6.764735 and MTTF = 987925

TABLE II

Input file	spec2.in		
mapping function			8
block size			4
writing strategy			Back
replacement algorithm			FIFO
Level	I	II	III
Memory size	4096	16384	32768
MTTF	1000000	90000000	120000000
fetch time	1 ns	10 ns	100 ns
read time	3 ns	20 ns	200 ns
write time	5 ns	50 ns	500 ns
miss penalty	10 ns	150 ns	300 ns
Trace			galgel_f2b

Our CDLR SPEC 2000 simulator, calculate the CDLR parameter, for the memory hierarchy. . In table II we present a memory hierarchy with the following parameters: the mapping function is 8 way set-associative, the size of a block is 4, the writing strategy is write back (1) and the replacement policy is FIFO (4). The parameters for the first level of the memory hierarchy are: the size of memory is 4096 bytes, the MTTF is 1000000s, the fetch time is 1ns, the read time is 3ns, write time is 5ns and miss penalty is 10ns. The parameters for the second level of the memory hierarchy are: the size of memory is 16384 bytes, the MTTF is 90000000s, the fetch time is 10ns, the read time is 20ns, write time is 50ns and miss penalty is 150ns. The parameters for the third level of the memory hierarchy are: the size of memory is 32768 bytes, the MTTF is 120000000s, the fetch time is 100ns, the read time is 200ns, write time is 300ns and miss penalty is 500ns.

The structure of the second output file (spec2.out), obtained after the computation of the CDLR SPEC 2000 with the second input file spec2.in (presented in Table II), is presented bellow :

----------------[LEVEL 1]--------------------------
MTTF Level : 1000000
Delay Average Level: 71613
Times Level: 12532177

----------------[LEVEL 2]--------------------------
MTTF Level : 90000000
Delay Average Level: 5822919
Times Level: 615085

----------------[LEVEL 3]--------------------------
MTTF Level : 120000000
Delay Average Level:49767785
Times Level: 139004
MTTF hierarcy: 980926
CDLR hierarcy: 45.259789

The CDLR parameter and the global MTTF for the memory hierarchy presented in table I are:
CDLR = 45.259789 and MTTF = 980926.

IV. CONCLUSIONS

Hierarchies of diverse storage levels have been analyzed extensively for their ability to achieve both good performance and low cost. This position paper argues that we should view hierarchies also as a way to achieve both good reliability and low overhead. We have suggested two metrics to use in evaluating the overall reliability of a reliability hierarchy: Mean Time To Data Loss (MTTDL) and Cache Data Loss Rate (CDLR). These and other metrics allow researchers to evaluate the reliability/performance tradeoffs quantitatively for new storage devices and hierarchies, especialy hierarchies with contain cache memories. It takes into account for the transfer policy between levels and is easy to compute based on our recurrent formula. Our model was validated by computer simulation. In an example we have demonstrated the usage of CDLR, which allow us to consider the tradeoff between reliability and performance.

The CDLR SPEC 2000 simulator also allows us to study the most suitable memory system for our needs before actually implementing it, or it simply helps us to simulate real systems in order to see their efficiency and compare results in an easy way.

Some of the parameters we can study with the simulator are: mapping, replacement policies, cache size (blocks in cache), number of cache sets (for set associative caches), number of words by block (block size).

ACKNOWLEDGMENT

*"University of Oradea", Department of Computers, Faculty of Electrical Engineering and Information Technology, Oradea, Romania,

**"Politehnica" University of Timişoara, Department of Computers, Timişoara, Romania.

***"University of Oradea", Department of Electrotechnics, Measurements and using of Electrical Energy, Faculty of Electrical Engineering and Information Technology, Oradea, Romania.

REFERENCES

[1] David A. Paterson, John L. Henessy – "Computer architecture. a quantitative approach", *Morgan Kaufmann Publishers*, Inc. 1990-1996.

[2] Peter M. Chen, David E. Lowell.,. Reliability hierarchies. *workshop on hot topics in operating systems*, 1999.

[3] Novac Ovidiu, Gordan M, Novac M., Data loss rate versus mean time to failure in memory hierarchies. *Proceedings of the CISSE'05,* University of Bridgeport, USA, 2005.

[4] Ooi, Y., M. Kashimura, H. Takeuchi, and E. Kawamura, Fault-tolerant architecture in a cache memory control LSI, *IEEE J. of Solid-State Circuits*, Vol. 27, No. 4, pp. 507-514, April 1992.

[5] Philip P. Shirvani and Edward J. McCluskey, "PADded cache: a new fault-tolerance technique for cache memories", *Computer Systems Laboratory, Technical Report* No. 00-802, Stanford University, Stanford, California, December 2000.

[6] Miguel A. Vega-Rodríguez, "SMPCache simulator". Available at http://arco.unex.es/smpcache.

A Group Mutual Exclusion Algorithm for Mobile Ad Hoc Networks

Ousmane THIARE
Department of Computer Science
LICP EA2175
University of Cergy-Pontoise-France
E-mail: othiare@dept-info.u-cergy.fr

Mohamed NAIMI
Department of Computer Science
LICP EA2175
University of Cergy-Pontoise-France
E-mail: naimi@dept-info.u-cergy.fr

Mourad GUEROUI
PRISM Lab
45, Avenue des Etats-Unis
78000 Versailles-France
E-mail: mogue@prism.uvsq.fr

***Abstract*- A mobile ad hoc network can be defined as a network that is spontaneously deployed and is independent of any static network. The network consist of mobile nodes[1] with wireless interfaces and has an arbitrary dynamic topology. The networks suffers from frequent link formation and disruption due to the mobility of the nodes. In this paper we present a token based algorithm for Group Mutual Exclusion in ad hoc mobile networks. The proposed algorithm is adapted from the RL algorithm in [1]. The algorithm requires nodes to communicate with only their current neighbors. The algorithm ensures the mutual exclusion, the bounded delay, and the concurrent entering properties.**

I. INTRODUCTION

A mobile ad hoc network can be defined as a network that is spontaneously deployed and is independent of any static network. The network consist of mobile nodes with wireless interfaces and has an arbitrary dynamic topology. The mobile nodes can communicate with only nodes in their transmission range and each one of them acts as router in routing data through the network. The network is characterized by frequent link formation and disruption due to the mobility of the nodes and hence any assumption about the topology of the network does not necessary hold.

Wireless links failure occur when nodes move so that they are no longer within tranmission range of each other. Likewise, wireless link formation occurs when nodes move so that they are again within transmission range of each other. In [1], an algorithm is proposed to solve the mutual exclusion problem for mobile ad hoc networks. The mutual exclusion problem is concerned with how to control nodes to enter the critical section to access a shared resource in a mutually exclusive way. The group mutual exclusion (*GME*) is a generalization of the mutual exclusion problem. In the *GME* problem, multiple resouces are shared among nodes. Nodes requesting to access the same shared resource may do so concurrently. However, if nodes compete to access different resources, only one of them can proceed.

In addition to the paper [1], there are papers proposed to solve mutual exclusion related problems for ad hoc networks. The paper [2] is proposed for solving the k-mutual exclusion problem, [4], for the leader election problem. There are several papers proposed to solve the *GME* problem for different system models. The papers [7][3][11] are designed for distributed message passing models, the paper [10], for self-stabilizing models. In this paper, we adapt the solution of [1] to solve the *GME* problem for mobile ad hoc networks.

The next section discusses related work. Section III presents our algorithm. We prove the algorithm correcteness in section IV. Conclusion and future work are offered in section V.

II. RELATED WORK

In [1], a token-based mutual exclusion algorithm, named *RL* (Reverse Link), for ad hoc networks is proposed. The *RL* algorithm takes the following assumptions on the mobile nodes and network:

1. the nodes have unique node identifiers,
2. node failures do not occur,
3. communication links are bidirectional and *FIFO*,
4. a link-level protocol ensures that each node is aware of the set of nodes with which it can currently directly communicate by providing indications of link formations and failures,
5. incipient link failure are detectable, providing reliable communication on a per-hop basis,
6. partitions of the networks do not occur, and
7. message delays obey the triangle inequality (i.e., messages that travel 1 hop will be received before messages sent at the same time that travel more than 1 hop).

The *RL* algorithm also assume that there is a unique token initially and utilize the partial reversal technique in [9] to maintain a token oriented *DAG* (directed acyclic graph). In the *RL* algorithm, when a node wishes to access the shared resource, it sends a request message along one of the communication link. Each node maintains a queue containing the identifiers of neighborings nodes from which it has received request for the token. The *RL* algorithm totally orders nodes so that the lowest-ordered nodes is always the token holder. Each node dynamically chooses its lowest-ordered neighbor as its outgoing link to the token holder. When a node detects a failure of an outgoing link and it is not the last outgoing one, it reroutes the request. If it is the last outgoing

[1] The terms processes and nodes will be used interchangeably throughout the paper.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 373–377.

link, there is no path to the token holder, so, it invokes a partial rearrangement of the *DAG* to find a new route. When a new link is detected, the two nodes concerned with this fact exchange message to achieve the necessary change in their outgoing and incoming links and to reroute eventually their requests. So, the partial rearrangement is called. The algorithm guarantees the safety and liveness property ([12] for the proof).

Now we present the scenario for the *GME* problem. Consider an ad hoc network consisting of *n* nodes and m shared resources. Nodes are assumed to cycle through a non-critical section (*NCS*), an waiting section (*Trying*), and a critical section (*CS*). A node i can access the shared resource only within the critical section. Every time when a node i wishes to access a shared resource S_i, node i moves from its *NCS* to the *Trying*, waiting for entering the *CS*. The *GME* problem [8] is concerned with how to design an algorithm satisfying the following property:

- **Mutual Exclusion**: If two distinct nodes, say *i* and *j*, are in the *CS* simultaneously, then $S_i = S_j$.
- **Bounded Delay**: If a node enter the *Trying* protocol, then it eventually enters the *CS*.
- **Concurrent Entering**: If there are some nodes requesting to access the same resource while no node is accessing a different resource, then all the requesting nodes can enter *CS* concurrently.

Note that this property is a trivial consequence of *Bounded Delay*, unless runs with nonterminating *CS* executions are admissible.

For now let us focus on executions where all request are for the same node. Joung's informal statement of *concurrent entering* was that (in such executions) nodes should be able not only to concurrently occupy the *CS* but to concurrently enter it without "unnecessary synchronisation". This means that (in such executions) nodes should not delay one another as they are *trying* to enter the *CS*. Concurrent occupancy ensures that a node i *trying* to enter the *CS* is not delayed by other nodes that have already entered the *CS*. It does not, however, prevent i from being delayed (for arbitrary long) by other nodes that are simultaneously *trying* to enter the *CS*.

III. Proposed Algorithm

A *DAG* is maintained on the physical wireless links of the network throughout algorithm execution as the result of a three-tuple, or triple, of integer representing the height of the node, as in [9]. Links are considered to be directed from nodes with higher height toward nodes with lower height, based on lexicographic ordering of the three tuple. A link between two nodes is *outgoing* at the higher height node and *incoming* at the lower height node. A total ordering on the height of nodes in the network is ensured because the last integer in the triple is the unique identifier of the node. For example, if the height at node 1 is (2,3,1) and the height at node 2 is (2,2,2), then the link between these nodes would be directed from node 1 to node 2. Initially at node 0, height is (0,0,0) and, for all $i \neq 0$, *i*'s height is initialized so that the directed links form a *DAG* in which every non token holder has a directed path to some token holder and every token holder. The lowest node is always the current token holder, making it a sink toward which all request are sent. In this section, we propose a distributed algorithm to solve the *GME* problem for ad hoc mobile network.

In this algorithm, we assume that all the nodes concurrently accessing the same resource terminate their tasks. The algorithm is assumed to execute in a system consisting of *n* nodes and *m* shared resources. Nodes are labeled as $0,1,...,n-1$, and resources are labeled as $0,1,...m-1$. We assume there is a unique token held by node 0 initially. The variable used in the algorithm for node i are listed below.

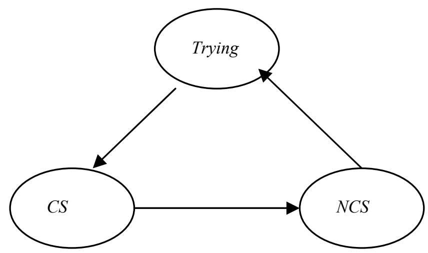


Fig. 1. States Process

- *status*: indicates whether node is the *Trying*, *CS*, or *NCS* section. Initially, *status*=*NCS*.
- *N*: the set of all nodes in direct wireless contact with node *i*. Initially, *N* contains all of node *i*'s neighbors.
- *Num*: counts the number of nodes within the critical section.
- *height*: a three-tuple (h_1, h_2, i) representing the height of node *i*. Links are considered to be directed from nodes with higher height toward nodes with lower height, based on lexicographic ordering. Initially at node 0, $height_0$=(0,0,0) and, for all $i \neq 0$, $height_i$ is initialized so that the directed links from a *DAG* where each node has a directed path to node 0.
- *Vect*: an array of tuples representing node *i*'s view of *height* of node *i*, $i \in N$. Initially, *Vect*[*i*]=*height* of node *i*. From *i*'s viewpoint, the link between *i* and *j* is *incoming* to node *i* if *Vect*[*j*]> $height_i$, and *outgoing* from node *i* if *Vect*[*j*]< $height_i$.
- *Leader*: a flag set to *true* if node holds the token and set to *false* otherwise. Initially, *Leader*=*true* if *i=0*, and *Leader*=*false* otherwise.
- *next*: indicates the location of the token from node *i*'s viewpoint. When node *i* holds the token, *next*=*i*, otherwise *next* is the node on an *outgoing* link. Initially, *next*=0 if *i*=0, and *next* is an *outgoing* neighbor otherwise.
- *Q*: a queue which contains request of neighbors. Operations on *Q* include *Enqueue*(), which enqueues

an item only if it is not already on Q, *Dequeue*() with the usual *FIFO* semantics, and *Delete*(), which remves a specified item from Q, regardless of its location. Initially, $Q = \varnothing$.

- *receivedLI*: boolean array indicating whether the height carrying message *LinkInfo* has been received from node *j*, to which a *Token* message was recently sent. Any height information received at node *i* from node *j* for which *receivedLI*[*j*] is false will not be recorded in *Vect*[*j*]. Initially, *receivedLI*[*j*]=*true* for all $j \in N$.
- *forming*[*j*]: boolean array set to *true* when link to node *j* has been detected as forming and reset to false when first *LinkInfo* message arrives from node *j*. Initially, *forming*[*j*]=*false* for all $j \in N$.
- *formHeight*[*j*]: an array storing the value of height of node *j*, $j \in N$, when new link to *j* is first detected. Initially, *formHeight*[*j*]=*height* for all $j \in N$.

The messages used in the algorithm are:

- *Request*(): when a node *i* whishes to enter the *CS* to access the resource *S*, it sends out *Request*() to the neighbor node indicated by *next*.
- *Okay*(): a message to inform nodes to access the resource *S* concurrently. There may be several *Okay*s in the system simultaneously.
- *Token*(): a message for node to enter the *CS*. The node with token is called the *Leader*.
- *Rel*(): a message for node *i* to release the resource S_i, it sends out *Rel*() to one of the neighbor node.
- *LinkInfo*(): a message used for nodes to exchange their height values with neighbors.

A. Pseudocode of the algorithm

When a node *i* requests access to the *CS*

1. $status \leftarrow Trying$
2. Enqueue(Q,*j*)
3. **If** (*not Leader*) **then**
4.   *If* ($|Q| = 1$) then *SendRequest()*
5. **Else** *SendTokenToNext*()

When node *i* release the *CS*

1. $status \leftarrow NCS$
2. **If** (*not Leader*) **then**
3.   *SendRel()*
4. **Else**
5.   **If** ($|Q| > 0$) **then** *SendTokenToNext*()

When Request(h) received at node *i* from node *j*

// *h* denotes *j*'s height when message was sent

1. **If** (*ReceivedLI*[*j*]) **then**
2.   *Vect*[*j*] $\leftarrow h$
3.   **If** (*height*<*Vect*[*j*]) then *Enqueue*(Q,*j*)
4.   **If** (*Leader*) **then**
5.   **If** (($status=NCS$) $\wedge$($|Q| > 0$))) **then**
6.   *SendTokenToNext*()
7.   **Else**
8.   **If** (*height*<*Vect*[*k*], $\forall k \in N$) **then**
9.   *RaiseHeight*()
10.   **Else If** ((Q=[*j*] $\vee$(($|Q| > 0$) $\wedge$($height < Vect[next]$)))
11.   *SendRequest*()

When *Rel*() is received at node i from node j

1. **If** (*Leader*) **then**
2.   **If** (*status*=*NCS*) **then**
3.   *SendTokenTonext*()
4.   **Else**
5.   *SendRel*()

When *Token*(*h*) received at node *i* from node *j*

1. *Leader* $\leftarrow$ *true*
2. *Vect*[*j*] $\leftarrow h$
3. $height.h_1 \leftarrow h.h_1$
4. $height.h_2 \leftarrow h.h_2 - 1$
5. Send *LinkInfo*(h_1, h, h_2, i)) to all $k \in N$
6. **If** ($|Q| > 0$)) **then** *SendTokenToNext*()

When *Okay*() received at node *i* from node *j*

1. Send *Okay*(S_j) to all $v \in Q$
2. **If** (($i \in Q$) $\wedge$ (*Resource*= S_j)) **then**
3.   *Num* $\leftarrow$ *Num*+1
4.   status $\leftarrow$ *CS*
5. **Else If** (($Q = [i]$) $\wedge$ ($Resource \neq S_j$)) **then**
6.   *Resource* $\leftarrow S_j$
7.   *Num* $\leftarrow 1$
8.   *SendRel*()

// Resource records the current resource identifier

When *LinkInfo*(h) received at node *i* from node *j*

1. $N \leftarrow N \cup \{j\}$
2. **If** ((*forming*[*j*]) $\wedge$($height \neq formHeight[j]$)) **then**
3.   Send *LinkInfo*(*height*) to *j*
4.   *forming*[*j*] $\leftarrow$ *false*
5. **If** *receivedLI*[*j*] **then** *Vect*[*j*] $\leftarrow h$
6. **Else If** (*Vect*[*j*]=*h)* **then** *receivedLI*[*j*]=*true*
7. **If** (($j \in Q$) $\wedge$ ($height > Vect[j]$)) **then** *Delete*(*Q*,*j*)
8. **If** (*Leader*) **then**
9.   **If** ((*height*<*Vect*[*k*], $\forall k \in N$) $\wedge$(*not Leader*)) **then**
10.   *RaiseHeight*()
11.   **Else If** (($|Q| > 0$) $\wedge$ ($height < Vect[next]$)) **then**
12.   *SendRequest*()

When failure of link to *j* is detected at node *i*

1. $N \leftarrow N - \{j\}$
2. *receivedLI*[*j*] $\leftarrow$ *true*

3. **If** ($j \in Q$) **then** *Delete*(*Q*,*j*)
4. **If** (*not Leader*) **then**
5. **If** ($height < Vect[k], \forall k \in N$) **then**
6. *RaiseHeight*()
7. **Else If** (($|Q| > 0$)) $\wedge (next \neq N)$) **then**
8. *SendRequest*()

When formation of link to *j* detected to node *i*
1. Send *LinkInfo*(*height*) to *j*
2. $forming[j] \leftarrow true$
3. $formHeight[j] \leftarrow height$

Procedure *SendRequest*()
1. $next \leftarrow l \in N : Vect[l] \leq Vect[j] \forall j \in N$
2. Send *Request*(*height*) to *next*

Procedure *SendTokenToNext*()
1. $next \leftarrow Dequeue(Q)$
2. **If** ($next \neq i$) **then**
3. $Leader \leftarrow false$
4. $Vect[next] \leftarrow (height.h_1, height.h_2 - 1, next)$
5 $receivedLI[next] \leftarrow false$
6. Send *Token*(*height*) to *next*
7. **If** ($|Q| > 0$)) **then** Send *Request*(*height*) to *next*
8. **Else**
9. $status \leftarrow CS$
10. Send $|Q|$ *Okay*s to all $v \in N \cap Q$

Procedure *RaiseHeight*()
1. $height.h_1 \leftarrow 1 + \min_{k \in N} \{Vect[k].h_1\}$
2. $S \leftarrow \{l \in N : Vect[l].h_1 = height.h_1\}$
3. **If** ($S \neq \emptyset$) **then** $height.h_2 \leftarrow 1 + \min_{l \in S} \{Vect[k].h_2\} - 1$
4. Send *LinkInfo*(*height*) to all $k \in N$
5. **For all** $k \in N$ such that $height > Vect[k]$ **do** Delete(*Q*,*k*)
6. **If** ($|Q| > 0$) **then** *SendRequest*()

Procedure *SendRel*()
1. **If** (*not Leader*) **then**
2. **If** ($height < Vect[k], \forall k \in N$) **then**
3. *RaiseHeight*()
4. $next \leftarrow l \in N : Vect[l] \leq Vect[j] \forall j \in N$
5. Send *Rel*() to *next*

IV. PROOF OF THE ALGORITHM

In this section we prove that the algorithm satisfies the three properties: mutual exclusion, the bounded delay, and the concurrent entering.

Theorem 1. The algorithm ensures mutual exclusion.

Proof 2. A node holding the token can enter the *CS* and then it informs all its requesting neighbors by sending out *Okay*s messages. When a neighbor node receives the message *Okay* information, it can enter the *CS* only if it requests for the same resource as the token holder. Because there is only a unique token, all nodes in the *CS* must access the same resource. Thus the property of mutual exclusion is satisfied.

Theorem 3. The algorithm ensures the property of concurrent entering.

Proof 4. A node holding the token can enter the *CS* and then it informs all its requesting neighbors by sending out *Okay*s messages. When a neighbor node receives the message *Okay* information, it can enter the CS only if it requests for the same resource as the token holder. Then all nodes can access the same resource as the token holder is currently accessing. Thus, the property of concurrent entering is satisfied.

Here, we prove that the algorithm satisfies the property of bounded delay by showing that if a node enter the *Trying* protocol, then it eventually enters the *CS* i.e., it holds the token. Since the height values of the nodes are totally ordered, the logical direction on links imparted by height values will eventually form a token oriented DAG. Before to prove the bounded delay property, we give the same lemma presented in [1].

Lemma 1. Once link failures cease, the logical direction on links imparted by height values will eventually form a token oriented DAG**.**

On the basis of Lemma 1, we prove that a requesting node holds token eventually.

Theorem 5. If link changes cease, then every request is eventually satisfied.

Proof 6. If a token holder i is in the *NCS*, it sends the token to the node *j* whose request is at the head of the queue. Node i then removes *j*'s request from the queue after sending the token. So, every node's request will eventually be at the head of the queue to have the opportunity to hold the token. By Lemma 1, every node's request has a path leading to the token holder. Thus, a requesting node holds the token eventually.

V. CONCLUSION AND FUTURE WORKS

We presented a token-based algorithm to solve the group mutual exclusion for mobile ad hoc networks. The algorithm is adapted from the RL algorithm in [1] and utilizes a concept used in [6] to detect that all nodes concurrently accessing the same resource have terminated their tasks. Simulations and message complexity is left as a future task.

REFERENCES

[1] J. Walter, J. Welch, and N. Vaidya. ''A Mutual Exclusion Algorithm for Ad Hoc Mobile Networks,'' *1998 Dial M for Mobility Workshop, Dallas TX*, 1998, 15 pgs.

[2] J. Walter, G. Cao, and M. Mohanty. ''A k-Mutual Exclusion Algorithm for Ad Hoc Wireless Networks,'' *Proceedings of the first annual Workshop on Principle of Mobile Computing (POMC 2001),* August, 2001.

[3] K.-P. Wu and Y.-J. Joung. ''Asynchronous group mutual exclusion in ring networks,'' *13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Processing (IPPS/SPDP '99)*, pages 539-543, 1999.

This page intentionally blank

[4] N. Malpani, J. L. Welch, and N. H. Vaidya. ''Leader Election Algorithms for Mobile Ad Hoc Networks,'' *Proc. Fourth International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications(PODC'99)*, pp. 96-103, 2000.

[5] Y.-C. Tseng. ''Detectint termination by Weight-Throwing in a Faulty Distributed System,'' *Journal of Parallel and Distributed Computing*, vol. 25, 1995, pp. 7-15.

[6] P. Keane and M. Moir. ''A simple local-spin group mutual exclusion algorithm.,'' *18th Annual ACM Symposium on Principles of Distributed Computing (PODC'99),* pages 23-32. ACM Press, 1999.

[7] Y.-J. Joung. ''The congenial talking philosophers problem in computer networks (extended abstract),'' *13th International Symposium on Distributed Computing -DISC'99)*, 1999.

[8] Y.-J. Joung. '' Asynchronous group mutual exclusion (extended abstract), '' *17th Annual ACM Symposium on Principles of Distributed Computing (PODC*), pages 51-60, 1998.

[9] E. Gafni and D. Bertsekas. ''Distributed algorithms for generating loop-free in networks with frequently changing topology.,'' *IEEE Transactions on Communication*, C-29(1):11-18, 1981.

[10] S. Cantarell and F. Petit. ''Self-Stabilizing Group Mutual Exclusion for Asynchronous Rings,'' *4th International Conference on Principles of Distributed Systems (OPODIS 2000)*, pages. 71-90, 2000.

[11] S. Cantarell, A. K. Datta, F. Petit, and V. Villain '' Token based group mutual exclusion for asynchronous rings,'' *21th International Conference on Distributed Computing Systems (ICDCS* (2001), pages 691-694, 2001.

[12] J.E. Walter and S. Kini. ''Mutual exclusion on multihop wireless networks,'' *Texas A\&M Univ., College Station, TX* 77843-3112, TR97-014, Dec 9, 1997.

An Extension of a CCM Environment with an Adaptive Planning Mechanism

Paweł Antoniewski, Łukasz Cygan, Jacek Cała, Krzysztof Zieliński
Department of Computer Science
AGH – University of Science and Technology
al. Mickiewicza 30, 30-059 Cracow, Poland

***Abstract*—The presented paper describes a promising approach of adaptive deployment in CORBA Component Model (CCM). Adaptation, as an essential concept of contemporary distributed systems, is used to improve planning of component deployment. There is described a concept and implementation of mechanisms which, using monitoring infrastructure and simple policy-based engine, allows improving execution efficiency. The presented tool, named CCMAD, is intended for use with OpenCCM platform, an open source Java implementation of CCM model. It is, however, expected that adapting CCMAD for any other CCM platform would not be troublesome.**

I. INTRODUCTION

Nowadays, adaptation of distributed applications to execution environment is very important issue in more and more complex computer systems sharing resources between many concurrent users. Efficient process of resources management in large computer systems is still very challenging task which draws attention of research community. This manifests in new research areas related to Next Generation Grid Initiative [12], Autonomic Computing proposed by IBM [10], Utility Computing started by SUN [14] and many others. All these initiatives recognize adaptation of applications as one of the most demanded features, which may influence almost every phase of application life-cycle.

This work discusses how deployment of component-based applications may benefit from adaptability. In this paper the broad vision of adaptability is limited to compositional adaptation [7] which determines how to modify structure of the software in response to changes in its execution environment. Despite that there are several other approaches providing application adaptation [2][3], component-based architectures are particularly convenient to introduce compositional adaptation. This is mainly due to distinct borders which define a component as well as inherent support for late binding [5]. In consequence, component-based software engineering allows effectively applying techniques such as redeployment and migration to improve application execution [4][6].

The proposed extension is targeted to CORBA Component Model (CCM) [8]. The CCM computational model represents one of the most advanced distributed component environments what fully justifies its selection as a subject of this study.

The defined CCM deployment process assumes that an application in form of assembled components is started over a target execution domain composed of connected computational nodes. The whole process consist of a few steps of which the planning is the most troublesome and complex. Deployment planning is a process of allocation of application component to the nodes and typically it relies on a static description of target domain resources.

The paper describes an extension of a CCM environment with mechanisms supporting adaptive application planning that take into account not only resource specification but also their current load and availability.

The text is structured as follows. In Sect. II there is briefly presented deployment of CCM applications. This creates background for the proposed concept of adaptive deployment mechanism presented in Sect. III. The constructed software tool, named CCMAD, supporting practical implementation of adaptive deployment process is described in more details in Sect. IV. The case study which compares efficiency of a testing application performance run by standard CCM deployment procedure with the same application deployed by CCMAD is presented in Sect. V. The paper is ended with conclusions.

II. PROCESS OF DEPLOYMENT OF CCM APPLICATIONS

The aim of deployment is to distribute, install, configure and connect software in a way which enables an application to execute in a target environment. OMG in *Deployment and Configuration of Component-based Distributed Applications specification* (D&C) [9] defines a course of this process for component-based applications which is further specialized in CORBA Component Model. Previously, however, CCM defined much more simplified approach to deployment of components.

According to D&C specification, deployment procedure consists of five steps: (1) installation in a software repository,

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 379–383.

(2) configuration of default package properties, (3) planning how and where the software will run, (4) preparation the target environment to execute the application, and finally, (5) launching the software in the target which includes instantiation, configuration and interconnection of the application components.

This paper turns attention to the planning phase, as the most complex part of the deployment process. Successful planning results in a *DeploymentPlan* which is a mapping of each component of an application to a selected node available in a target execution domain. In this paper we show that effectiveness of the application execution might strongly depend on the way the components are arranged in the domain.

The main issue of planning the deployment is to match component requirements to suitable resource satisfier properties of the execution environment. Using D&C specification, developers have considerable degree of freedom in expressing component requirements and target domain properties. Both, *Requirement*s and resource properties in form of *RequirementSatisfier*s are composed of a list of name-value pairs and a type as a string of characters. The satisfiers have also additional information about their kind (e.g. capacity, quantity) as well as whether they are dynamic or static attributes (listing below shows requirement satisfier of a sample resource). This very general approach allows defining potentially any kind of a resource or requirement but also raises the issue of a common language to express them conveniently such that they are comparable and might be matched against each other.

Listing 1. An example of requirement satisfier of a CPU resource

```
<resource>
  <name>HostA CPU</name>
  <resourceType>cpu</resourceType>

  <property>
    <name>number</name>
    <kind>Quantity</kind>
    <value>
      <type><kind>tk_ulong</kind></type>
      <value><ulong>1</ulong></value>
    </value>
  </property>
  <property>...</property>
  ...
</resource>
```

Apart from the new approach of deployment defined in D&C, previously, CORBA Component Model introduced much more simplified way of describing application components. There were no means to define requirements of a component, there were also no means to describe a target environment. As a consequence, previous CCM specification did not undertake planning phase at all providing no means to describe a deployment plan.

Nevertheless, in order to run a component application in distributed environments, platforms implemented according to the older approach were forced to supply solution to this deficiency. This article is based on OpenCCM platform [13], to the best of our knowledge, the most advanced freely available Java implementation of CORBA Component Model. OpenCCM, based on the previous deployment procedure, extends the specification by defining an additional element of Component Assembly Descriptor (CAD). Therefore, a developer is able to assign an eligible destination node to a *homeplacement*, *hostcollocation*, or *processcollocation*.

A home placement element is a standard measure to describe a group of component instances managed by a single common component home, whereas process and host collocations allow defining bonds between two or more home placements. This binding between the placements enforces a home to be run in the same process or on the same machine respectively. Listing below presents an excerpt from a sample CAD file with the OpenCCM extension – a *destination* element.

Given such supplemented assembly descriptor, which now

Listing 2. The destination element – the extension to original Component Assembly Descriptor file

```
<componentassembly id="myApplication">
...
<partitioning>
  <homeplacement cardinality="1"
                 id="ConsumerHome">
    <componentinstantiation id="aComponent">
      ...
    </componentinstantiation>
    ...
    <destination>
      <installation type="component installation">
        <findby>
          <namingservice
                        name="ComponentServer1" />
        </findby>
      </installation>
    </destination>
  </homeplacement>
  ...
</partitioning>
<connection>
  ...
</connection>
```

might be perceived as a simple deployment plan, OpenCCM deployment tools are able to distribute, install, configure and run application components on desirable component servers running on particular nodes.

III. CONCEPT OF ADAPTIVE PLANNING MECHANISMS FOR CCM

The analysis of the deployment procedure presented in the previous section leads to the conclusion that two most important issues which a planner should perform are: matching the requirements of application components to the resources in a target execution environment and then distribution of the components among the corresponding nodes.

The former problem is connected with heterogeneity of a destination environment. It results in a set of nodes generated for each component in such a way that all of them are able to host that component. Here, in this article, we intentionally skip this stage assuming that every node in the target domain can host any component of the application. This allows us to focus on the latter more interesting issue – the component distribution.

If matching requirements does not provide unambiguous answer where to place a component, there should be found other criteria which allow the planner to reach the decision. We propose to use an adaptive technique to aid this process. Figure 1 illustrates where the planner is located in the process of deployment. Adaptiveness, in this context, is achieved by realization of a strategy or strategies expressed with user-provided policies. The policies may operate on a broad range of attributes supplied to the CCMAD such as a description of application composition and resource information gathered by a monitoring system. For example, a policy may take into account total number of components to be deployed, temporary CPU load, amount of free memory, type of an operating system on a host machine etc. all this giving basis for different distribution patterns. As shown later in this paper, even simple strategies may have great impact on application organization and execution effectiveness.

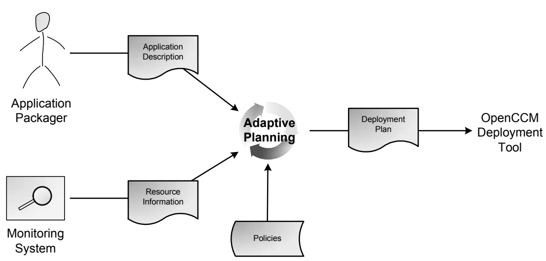


Fig. 1. Flow of information during planning with CCMAD

It is important to mention that, in our solution, the planner is supported by user-provided policies. This ensures better flexibility as the user may freely express their planning goals and distribution strategies which are intended to be achieved. Moreover, external policies make the solution less dependent on a kind and type of information provided by a monitoring system and allows using it in a broader range of environments.

IV. IMPLEMENTATION OF ADAPTIVE DEPLOYMENT PROCESS

Despite that implementation of CCMAD tool is substantially depended on OpenCCM platform, to present the proposed solution of adaptive planning it is required to answer the following general questions:

1) How to get information about available target domain nodes?
2) How to monitor most important attributes of the nodes found in the previous step?
3) How to process gathered information to get distribution scenarios consistent with desired requirements?
4) How to use the scenarios obtained in step 3 to influence deployment of an application?

Gaining the information of where are the potential nodes able to host the application is an inevitable step on the way to run any distributed software. CCMAD helps to automate this task by making use of the *Distributed Computing Infrastructure* (DCI) provided with OpenCCM. DCI requires all the nodes in the domain to run a *NodeManager* and one or more *ComponentServers* which register themselves in a naming service. Having access to the naming service, CCMAD is able to acquire the list of all potential hosts for components.

Subsequently, this list of hosts is used to gather monitoring information. What kind of data is collected and how frequently depends, however, on the adopted policy. In the simplest case e.g. with round-robin deployment policy there is no need to get any monitoring information because components are assigned to nodes basing merely on the number of components to be deployed and the number of available component servers. On the other hand, more advanced policies may take into account static and dynamic attributes provided by the monitoring system.

To monitor an execution domain CCMAD uses JIMS monitoring system which proved useful for many applications and especially in grid monitoring [1]. JIMS allows gauging the most important dynamic attributes of an operating system such as CPU load, memory consumption, and storage availability what plays the pivotal role for adaptive planning. This information may further be accessed by the provided policies.

The core of the CCMAD is an adaptive planner which, using provided information and policies, generates an appropriate plan. A policy is a set of rules specified by an end-user in two equivalent forms: (1) as a Java class, or (2) a as Groovy script [11], if more concise and simpler form is preferred. To develop a policy a policy creator uses as input a list of home placements to be deployed and a list of nodes available in a target domain. Additionally, this information may be completed with monitoring data received from JIMS system giving chance to program desired rules of distribution. Having the policy the planner is able to produce an appropriate deployment plan.

The last issue is to make use of the plan provided in the previous step. Since OpenCCM platform is based on the older specification of CCM deployment the only reasonable way to introduce the generated plan to the platform is to change Component Assembly Descriptor extracted from the application description. As said in Sect. II, OpenCCM extends the original CAD file definition with an additional *destination* element which determines a component server where a home placement or collocation are going to be instantiated. CCMAD applies the plan by supplementing a CAD file with missing destination information. This results in a well prepared assembly descriptor used by OpenCCM deployment tools to install the application.

It is worth to note that without CCMAD distribution of components has to be performed manually by an application packager who has to envisage how many nodes there are in the domain and where is the best place to instantiate the components. CCMAD automates this process and, moreover, adapts the distribution to the most recent resource availability.

V. Performance evaluation

To perform evaluation test of usability and gain provided by the proposed approach there was selected a simple processing application. The model of application, presented in Fig. 2, consists of a Manager component and several Workers. Manager dispatches some tasks to Workers which perform computation and return back their results. In this evaluation we will measure time required to accomplish a job sent by Manager in different testing scenarios.

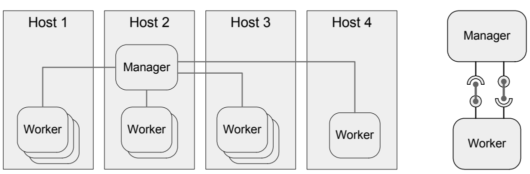


Fig. 2. Model of the testing application and the type of connections between manager and worker components

The tests were performed on a group of 4 identical machines each with two Intel IA-64 1.3 GHz CPUs. To simulate different testing scenarios on each machine there was running a load generator consuming a portion of CPU resources. The exact usage patterns are presented in Table I.

Each test scenario was executed with one Manager and different number of Worker components (from 5 to 50). Moreover, for each test scenario with different number of components two experiments were performed: one with CCMAD adaptive planning and one with manual component distribution where all components were evenly spread across the available nodes. The policy used by CCMAD to deploy Worker components was simple and allocated the number of components to a host which was proportional to residual computational power of a node.

As shown in Fig. 3, even such simple policy gives significant gains in application performance. In all scenarios (except 1 when CCMAD distributes components evenly what is the best mapping) there is reduction roughly about 5–60% in execution time (see Table II). These results are strong motivation for further research in this area.

TABLE I
CPU Consumption Patterns for Different Test Scenarios

	Host 1	Host 2	Host 3	Host4
Test Scenario 1	0%	0%	0%	0%
Test Scenario 2	0%	0%	50%	50%
Test Scenario 3	0%	20%	80%	80%
Test Scenario 4	0%	0%	100%	0%
Test Scenario 5	0%	30%	30%	30%

TABLE II
A Percent Gain of Application Performance for Different Test Scenarios and Component Number

Test Scenario	Number of Components 10	20	30	50
1	0	0	0	0
2	26	17	22	13
3	28	12	19	19
4	56	58	60	52
5	3	2	8	9

VI. Conclusions

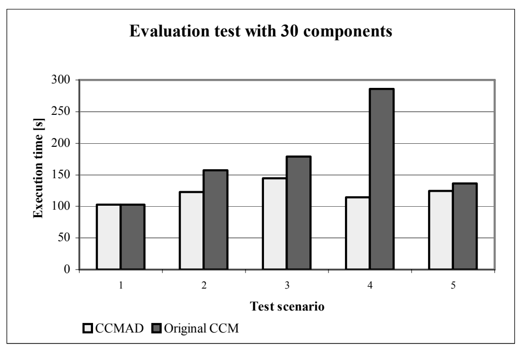


Fig. 3. A gain achieved using CCMAD tool in different test scenarios

The paper presents some preliminary work in the area of adaptive deployment. The proposed solution is the beginning and may be further developed in many research directions such as an automatic domain analysis and description, more advanced policy definition and processing.

Nevertheless, the presented study shows two very important facts: (1) planning of an application deployment should be supported by adaptive mechanisms in order to use available resources more effectively; (2) even simple policy together with accurate monitoring system can provide substantial gain in performance of distributed component applications.

The study shows also that CAD descriptor, which specifies architecture of an application, does not convey information

This page intentionally blank

that may be used by more sophisticated deployment policies. Only package configuration created according to D&C specification will allow more detailed description taking into account resource requirements of an application. D&C, however, does not provide means to include monitoring or QoS policies into an application description, and exploiting this concepts is the further step of the presented research.

REFERENCES

[1] K. Bałos and K. Zieliński, "JIMS – the uniform approach to gird infrastructure and application monitoring," in *Proceedings of Cracow Grid Workshop 2004*, pp. 160–167, 2004.

[2] Jérémy Buisson, *Performance and practicability of dynamic adaptation for parallel computing: an experience feedback from Dynaco*. Technical report, IRISA/INRIA Rennes, February 2006. Available at: http://hal.inria.fr./inria-00001087.

[3] B. Ensink, J. Stanley, and V. Adve, "Program control language: A programming language for adaptive distributed applications," *Journal of Parallel and Distributed Computing*, October 2002.

[4] A. Fuggeta, G.P. Picco, and G. Vigna. Understanding code mobility. *IEEE Transactions on Software Engineering*, (5):342–361, May 1998.

[5] D. Garlan, B. Schmerl, and J. Chang. "Using gauges for architecture-based monitoring and adaptation," in *Proceedings of the Working Conference on Complex and Dynamic Systems Architecture*, December 2001.

[6] S. Malek, M. Mikic-Rakic, and N. Medvidovic. "A decentralized redeployment algorithm for improving the availability of distributed system," in *Component Deployment 2005*, pp. 99–114, 2005.

[7] P.K. McKinley, S.M. Sadjadi, E.P. Kasten et al. "Composing adaptive software," *Computer*, pp. 56-64, July 2004.

[8] Object Management Group, Inc. *CORBA component model specification*, version 4.0, April 2006.

[9] Object Management Group, Inc. *Deployment and configuration of component-based distributed applications specification*, version 4.0, April 2006.

[10] Autonomic Computing Initiative. Web site at: http://www-03.ibm.com/autonomic.

[11] Groovy language. Web site at: http://groovy.codehaus.org.

[12] Nextgrid: Architecture for next generation girds. Web site at: http://www.nextgrid.org.

[13] OpenCCM – the open CORBA components model platform. Web site at: http://openccm.objectweb.org.

[14] Utility computing initiative. Web site at http://www.sun.com/service/sungrid/overview.jsp.

Cross-Trial Query System for Cancer Clinical Trials

Radu Calinescu, Steve Harris, Jeremy Gibbons and Jim Davies
Computing Laboratory, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK

Abstract **Data sharing represents one of the key objectives and major challenges of today's cancer research. CancerGrid, a consortium of clinicians, cancer researchers, computational biologists and software engineers from leading UK institutions, is developing open-standards cancer informatics addressing this challenge. The CancerGrid solution involves the representation of a widely accepted clinical trials model in controlled vocabulary and common data elements (CDEs) as the enabling factor for cancer data sharing. This paper describes a cancer data query system that supports data sharing across CancerGrid-compliant clinical trial boundaries. The formal specification of the query system allows the model-driven development of a flexible, web-based interface that cancer researchers with limited IT experience can use to identify and query common data across multiple clinical trials.**

I. INTRODUCTION

CancerGrid [1] is a project that brings together clinicians, cancer researchers, bioinformaticians and software engineers from leading UK institutions. The project uses a generic clinical trials model [2] based on controlled vocabulary and common data elements (CDEs) [3] for the design, execution and analysis of cancer clinical trials. The advantage of this approach is twofold. Firstly, it makes possible the model-based development of open-standards IT systems for clinical trial execution [4]. Secondly, the CancerGrid approach enables data sharing across cancer clinical trial boundaries. The latter capability is demonstrated by the cross-trial query system presented in this paper.

The remainder of the paper is organised as follows. After a review of related work in the next section, Section III uses Z notation [5] to formally define common data elements. Section IV shows how CDE sets are associated to trial events to create a *trial design*, i.e., a full specification of a cancer clinical trial. Section V describes the cross-trial query model, and its use of the CDEs associated with the same execution stage (e.g., patient registration or follow-up) of multiple trials.

A proof-of-concept system that implements the cross-trial query model is presented in Section VI. The system allows cancer researchers with limited IT experience (e.g., clinicians and statisticians) to easily identify common data across multiple clinical trials, and to build queries targeted at these data using a familiar interface.

Section VII discusses two possible extensions of the query model. First, the grouping of CDEs associated with different trial execution stages is considered as a way of making queries less restrictive. Second, a solution to the generation of queries compliant with the security constraints specific to clinical trials is investigated. Section VIII concludes the paper with an overview of the query system, and an analysis of future work directions.

II. RELATED WORK

The query of data from multiple sources has been an important research topic for the last two decades. Generic approaches for querying multiple information sources were proposed [6, 7, 8] that use a model of a problem domain to devise global query systems. The approach in [6] requires the user to build a semantic *domain model* as well as a model of each database and knowledge base used as an information source. Therefore, this solution is appropriate only for users with expertise in both data modelling and the target problem domain. Similar approaches are described in [7, 8], where sophisticated techniques are used to create a "metadatabase" [7] or a "reference data model" [8] that are then employed to generate the global query. Unlike these approaches that address the query of heterogeneous data sources, our query system takes advantage of the homogeneity of data across cancer clinical trials to hide most of the complexity of a cross-trial query. Implementations of this system can therefore be used directly by cancer researchers with limited data modelling expertise.

In the cancer research area, the US cancer Biomedical Informatics Grid (caBIG) project [9] models clinical trials [10] and has cancer data sharing as one of its primary objectives. Their caCORE software development kit [11] provides building blocks for many software components employed in cancer research. The inclusion of multiple data source querying in a proprietary language (i.e., the caBIG Query Language) is planned for the next release of the kit. While this will provide the functionality required to implement a system for querying multiple cancer data sources, the query system presented in this paper allows the automatic generation of a complete query form ready for immediate use by clinicians and statisticians.

Other medical projects such as VOTES [12] and PRATA [13] are concerned with the integration of data from multiple, distributed databases. The VOTES system [12] is concerned with the integration of distributed medical data pertaining to the same patient, so candidate patients for new clinical trials can be identified easily. The query forms used by the VOTES portal resemble those from the prototype implementation of the query system introduced in this paper, however they are encoded manually by software developers familiar with the internal structure of the data sources. The PRATA system [13] addresses the XML integration of data extracted from multiple, distributed databases. The integration and visualisation of the data is based on a user-specified XML schema that requires inside knowledge of the data sources. On the contrary, the CancerGrid query forms are model-based, and provide information to guide user querying rather than relying on the users for this knowledge.

III. COMMON DATA ELEMENTS

The consistent use of a controlled vocabulary (i.e., a list of explicitly enumerated terms managed by a vocabulary

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 385–390.

registration authority) is key to sharing data between projects in any field of research. This is particularly relevant to cancer research, where tremendous human and financial resources are often employed for the generation of relatively small amounts of data [1]. The ability to analyse these data across multiple clinical trials is crucial to reaching statistically relevant conclusions.

The CancerGrid project is addressing this important requirement by basing its clinical trials model [2] on the use of thesauri—collections of controlled vocabulary terms and their relationships, and common data elements—controlled sets of cancer concepts and measurements. A common data element [3] is defined in terms of several basic types:

- *CdeID*, the set of all common data element identifiers. CDE identifiers are used to uniquely refer to specific CDEs.
- *CdeType*, the set of all types that CDE values may have. Typically, any XML schema simple type is allowed.
- *CdeInfo*, the actual details of the CDE, including a name and a description.

These basic types are summarised below using Z notation [5]:

$$[CdeID, CdeType, CdeInfo], \tag{1}$$

and the common data element type can be specified as:

$$\begin{array}{|l}\hline \textit{Cde} \\ id : CdeID \\ valueDomain : CdeType \\ info : CdeInfo \\ \hline\end{array} \tag{2}$$

Common data elements used to model data in a specific research field are maintained in a CDE (or metadata) repository for that area of research:

$$\begin{array}{|l}\hline \textit{CdeRepository} \\ cdeSet : \mathbb{P}\, Cde \\ \hline \forall x, y : cdeSet \bullet x.id = y.id \Rightarrow x = y \\ \hline\end{array} \tag{3}$$

IV. Clinical Trial Events and Trial Designs

Clinical trial data are generated during the execution of a trial as a result of a number of trial events, each of which corresponds to a stage in the execution of the clinical trial. For instance, clinical and personal patient data are collected during the *registration* stage, treatments are allocated in the *randomisation* stage, and periodical *follow-up* data collection is performed to assess response to treatment. The complete set of trial events in the CancerGrid trial model is given below:

$$\begin{aligned} TrialEvent ::= {} & registration \mid eligibility \mid randomisation \mid \\ & onStudy \mid treatment \mid offStudy \mid response \mid \\ & followUp \mid adverseEffect \end{aligned} \tag{4}$$

Clinicians gather the data corresponding to the trial events by filling in case report forms that comprise CDEs drawn from the cancer CDE repository [3],

$$\mid\ cancerCdeRep : CdeRepository. \tag{5}$$

For the purpose of data analysis, a clinical trial is composed of a set of trial events, each of which is associated with a set of common data elements [2]. This is defined by the *TrialDesign* specification below:

$$\begin{array}{|l}\hline \textit{TrialDesign} \\ events : \mathbb{P}\, TrialEvent \\ eventCdeSet : TrialEvent \nrightarrow \mathbb{P}\, cancerCdeRep.cdeSet \\ \hline \operatorname{dom} eventCdeSet = events \\ \hline\end{array} \tag{6}$$

To give an example of a clinical trial design, consider the following common data elements:

$$\begin{array}{|l} NodalStatus, AdjuvantRadiotherapy, Her2Level, \\ ECOGStatus, InvasiveCarcinoma, TumorResectionStatus, \\ DiseaseStage, AdjuvantChemotherapyIndication, \\ PatientFitness, BoneMarrowHepaticRenalFunction, \\ InformedConsent, NoPreviousTherapy, KnownRadiotherapy, \\ LastSurgeryDate, NoPreviousMalignancy, \\ NotPregnantLactating, PatientNameInitials, \\ PatientBirthDate, TissueSubstudyConsent, \\ QualityOfLifeSubstudyConsent, ProgesteroneReceptorStatus, \\ OestrogenReceptorStatus : cancerCdeRep.cdeSet \end{array} \tag{7}$$

that are associated with three of the trial events for the tAnGo clinical trial [14]:

$$\begin{array}{|l} tAnGo : TrialDesign \\ \hline \{registration, eligibility, randomisation\} \subset tAnGo.events \\ tAnGo.eventCdeSet\ registration = \\ \quad \{TissueSubstudyConsent, QualityOfLifeSubstudyConsent, \\ \quad ProgesteroneReceptorStatus, OestrogenReceptorStatus\}. \\ tAnGo.eventCdeSet\ eligibility = \\ \quad \{InvasiveCarcinoma, TumorResectionStatus, \\ \quad DiseaseStage, AdjuvantChemotherapyIndication, \\ \quad PatientFitness, BoneMarrowHepaticRenalFunction, \\ \quad InformedConsent, NoPreviousTherapy, \\ \quad KnownRadiotherapy, LastSurgeryDate, \\ \quad NoPreviousMalignancy, NotPregnantLactating\} \\ tAnGo.eventCdeSet\ randomisation = \\ \quad \{NodalStatus, AdjuvantRadiotherapy, Her2Level, \\ \quad ECOGStatus\} \end{array} \tag{8}$$

The common data elements used to register *tAnGo* participants, to establish their eligibility, and to stratify the allocation of treatments for the eligible participants (i.e., the trial *randomisation* [15]) are explicitly specified in the *tAnGo* trial design. Note that the complete trial design for tAnGo comprises all of the trial events defined in (4), however for the sake of brevity only three of these are presented above.

V. Clinical Trial Queries

Having introduced the data components of a CancerGrid clinical trial in the previous section, we will now define the cross-trial queries for sharing data among clinical trials using the same CDE repository. A number of comparison operators are used to build the query:

$$\begin{array}{l} ComparisonOp ::= hasAnyValue \mid isEqualTo \mid \\ \quad isNotEqualTo \mid isLessThan \mid isGreaterThan \mid \\ \quad isLessThanOrEqualTo \mid isGreaterThanOrEqualTo. \end{array} \tag{9}$$

Each *CdeType* type that CDEs can draw their values from is associated with a well-defined set of these operators:

$$cdeComparisonOp : CdeType \rightarrow \mathbb{P}\ ComparisonOp. \tag{10}$$

For our purpose, a query element comprises a CDE and a comparison operator that is relevant for its value domain:

$$\begin{array}{|l} \textit{QueryElement} \\ \hline cde : Cde \\ op : ComparisonOp \\ \hline op \in cdeComparisonOp\ cde.valueDomain \end{array} \tag{11}$$

The query system described in this section is event based, namely we are interested in identifying CDEs that are associated with the same trial event in all clinical trials involved in the query. This approach is consistent with the cancer researchers' need to analyse data from patients with similar characteristics at the same stage of their treatment. For instance, it makes sense to group patient data collected prior to the commencement of treatment for trials where the treatment varies, because this will avoid the confounding effects of the different treatments under study. However, comparisons at later time points may prove less useful.

Given a set of clinical trials *trialSet*, the query system specifies its query terms as a mapping from trial events to sets of query elements:

$$\begin{array}{|l} \textit{TrialQuery} \\ \hline trialSet : \mathbb{P}\ TrialDesign \\ queryTerms : TrialEvent \rightarrowtail \mathbb{P}\ QueryElement \\ \hline \mathrm{dom}\ queryTerms = \bigcap\{t : trialSet \bullet t.events\} \\ \forall\, e : \mathrm{dom}\ queryTerms \bullet \{qt : queryTerms\ e \bullet qt.cde\} = \\ \quad \bigcap\{t : trialSet \bullet t.eventCdeSet\ e\} \\ \forall\, e : \mathrm{dom}\ queryTerms \bullet \forall\, qt1, qt2 : queryTerms\ e \bullet \\ \quad qt1 \neq qt2 \Rightarrow qt1.cde \neq qt2.cde \end{array} \tag{12}$$

The first constraint in the query definition requires that the trial events involved in the query must be part of all considered trials.[1] The last two constraints state that the query terms for these events comprise precisely one *QueryElement*[2] for every CDE that the event is associated with in each of the queried trials. Notice that the *TrialQuery* specification in (12) does not place any restriction on the comparison operators that are part of the *QueryElement* items associated with trial events. The only such constraint is specified by the definition of a *QueryElement* (11), i.e., these operators must be appropriate for the CDEs they relate to.

To illustrate the application of the trial query, consider the NEAT clinical trial [16], which uses several additional CDEs alongside those introduced in the previous section:

$$\begin{array}{|l} RadiotherapyTiming, TumorSize, TumorGrade, \\ CyclophospamidePlan, MenopausalStatus, \\ TamoxifenPlan : cancerCdeRep.cdeSet \\ \hline Neat : TrialDesign \\ \hline \{registration, eligibility, randomisation\} \subset Neat.events \\ Neat.eventCdeSet\ registration = \\ \quad \{QualityOfLifeSubstudyConsent, \\ \quad OestrogenReceptorStatus, TumorSize, \\ \quad TumorGrade, ECOGStatus, CyclophospamidePlan, \\ \quad MenopausalStatus, TamoxifenPlan\} \\ Neat.eventCdeSet\ eligibility = \\ \quad \{InvasiveCarcinoma, DiseaseStage, \\ \quad TumorResectionStatus, AdjuvantChemotherapyIndication, \\ \quad PatientFitness, InformedConsent, \\ \quad BoneMarrowHepaticRenalFunction, \\ \quad NoPreviousMalignancy, NotPregnantLactating\} \\ Neat.eventCdeSet\ randomisation = \\ \quad \{NodalStatus, RadiotherapyTiming\} \end{array} \tag{13}$$

An event-based query across the tAnGo trial in (8) and the NEAT trial above is then given by:

$$\begin{array}{|l} tAnGoNeatQuery : TrialQuery \\ \hline tAnGoNeatQuery.trialSet = \{tAnGo, Neat\}. \end{array} \tag{14}$$

According to the definition of a *TrialQuery* in (12),

$$\begin{array}{c} \{registration, eligibility, randomisation\} \subseteq \\ \mathrm{dom}\ tAnGoNeatQuery.queryTerms \end{array} \tag{15}$$

since all these trial events appear in both *tAnGo* and *Neat*. The CDE sets that are part of the *queryTerms* (12) for the three trial events are:

$$\begin{array}{l} \{qt : tAnGoNeatQuery.queryTerms\ randomisation \bullet qt.cde\} = \\ \quad \{NodalStatus\} \\ \{qt : tAnGoNeatQuery.queryTerms\ eligibility \bullet qt.cde\} = \\ \quad \{InvasiveCarcinoma, TumorResectionStatus, \\ \quad DiseaseStage, AdjuvantChemotherapyIndication, \\ \quad PatientFitness, BoneMarrowHepaticRenalFunction, \\ \quad InformedConsent, NoPreviousMalignancy, \\ \quad NotPregnantLactating\} \\ \{qt : tAnGoNeatQuery.queryTerms\ registration \bullet qt.cde\} = \\ \quad \{QualityOfLifeSubstudyConsent, \\ \quad OestrogenReceptorStatus\}. \end{array} \tag{16}$$

Appropriate comparison operators are associated with each of these CDEs in the above *TrialQuery* instance, e.g.,

$$\begin{array}{l} tAnGoNeatQuery.queryTerms\ randomisation = \\ \qquad \{\langle\!| cde \leadsto NodalStatus, op \leadsto isEqualTo |\!\rangle\} \end{array} \tag{17}$$

Prior to being executed, a *TrialQuery* instance such as *tAnGoNeatQuery* needs to be parameterised by a set of values from the value domains of all CDEs in the query terms. For the *tAnGoNeatQuery* query term in (17) for instance, *op* was chosen to be *isEqualTo*, so the *NodalStatus* value of interest will need to be specified in an implementation of the query framework. This part of the query is not modelled here, however details are provided in the next section that presents a proof-of-concept realisation of the system.

---

[1] This simplifying assumption is relaxed in Section VII, which proposes a generalisation of the basic cross-trial query in (12).

[2] Although each of the *queryTerms* consists of a single *QueryElement*, the query model can be extended easily to handle terms defined as logical expressions of multiple *QueryElements* that refer to the same CDE.

VI. Case Study

The cross-trial query system introduced in the previous section was applied to two of the primary CancerGrid clinical trials, tAnGo [14] and NEAT [16]. The Microsoft ASP.NET platform [17] was chosen for the implementation of the query system prototype. This choice was motivated by a range of factors including customer acceptance, simplicity, and the ability to reuse existing model-based web form generators that CancerGrid developed for the automatic generation of service-oriented architectures (SOAs) for clinical trial execution [4].

The reuse of the CancerGrid SOA generation components ensures that changes and enhancements to the query system can be easily converted into query web forms. Additionally, this approach provides a query user interface that clinical trial personnel will already be familiar with after having filled in similar forms during the actual execution of the trial.

Fig.1 shows the query web form generated for the two clinical trials. As shown at the top of Fig. 1, in the degenerate case when an empty *trialSet* is used, the *queryTerms* set is itself empty:

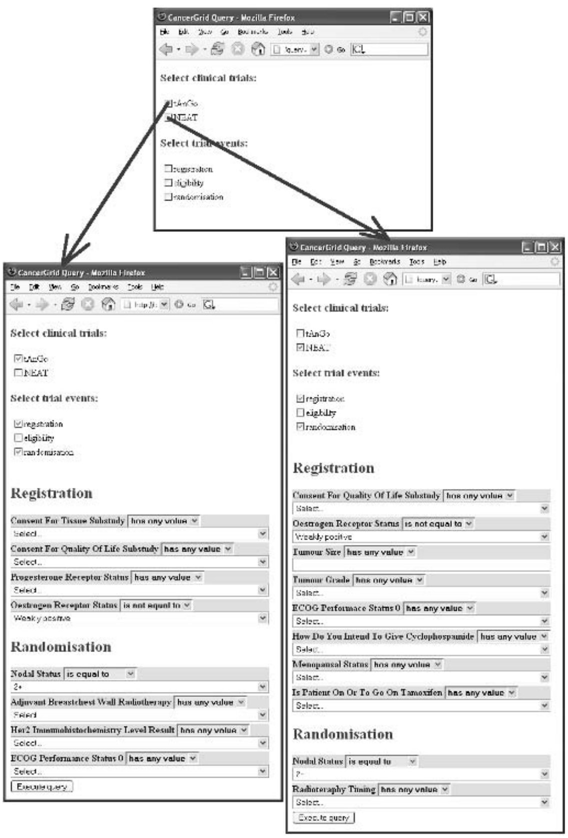


Fig. 1. Single-trial queries as a special case of the cross-trial query system. The familiar, easy-to-use web form allows cancer researchers with limited IT experience to identify and query common data across multiple clinical trials, and have their query automatically encoded into a set of dedicated SQL statements, one for each of the trials.

$$\begin{array}{|l} emptyQuery : TrialQuery \\ \hline emptyQuery.trialSet = \emptyset \end{array} \tag{18}$$

By adding either trial to the *trialSet*, an ordinary, single-trial query system is obtained (Fig. 1), e.g.,

$$\begin{array}{|l} tAnGoQuery : TrialQuery \\ \hline tAnGoQuery.trialSet = \{tAnGo\}. \end{array} \tag{19}$$

Although this use case does not involve any data sharing, the ability to perform ordinary trial queries using the same system is useful. The power of the cross-trial query technique is put to use by including both the *tAnGo* and the *Neat* clinical trials into the *trialSet*, as defined by *tAnGoNeatQuery* in (14) and illustrated in Fig. 2.

VII. Query System Extensions

A. Trial event combining

Common data elements used in different clinical trials are not necessarily associated with the same trial event in each of these trials. An example is the *ECOGStatus* CDE defined in (7), which is used by both trials considered in the previous sections:

$$ECOGStatus \in tAnGo.eventCdeSet\ randomisation \tag{20}$$

but

$$ECOGStatus \in Neat.eventCdeSet\ registration, \tag{21}$$

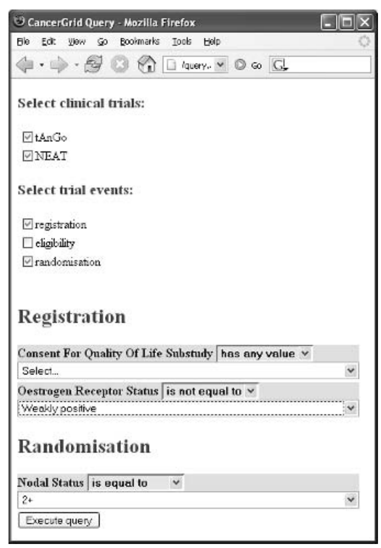


Fig. 2. The *tAnGoNeatQuery* cross-trial query, with two trial events selected. The sets of query terms for the randomisation and registration trial events are $tAnGoNeatQuery.queryTerms\ randomisation = \{\langle\!| cde \leadsto NodalStatus, op \leadsto isEqualTo |\!\rangle\}$ and $tAnGoNeatQuery.queryTerms\ registration = \{\langle\!| cde \leadsto QualityOfLifeSubstudyConsent, op \leadsto hasAnyValue |\!\rangle, \langle\!| cde \leadsto OestrogenReceptorStatus, op \leadsto isNotEqualTo |\!\rangle\}$, respectively.

In both cases the measurement is taken prior to patients joining the clinical trial, so the ability to perform queries including query terms for such CDEs is very important.

Relaxing the requirement that cross-trial queries are event-based (in the sense described in the previous section) opens up a whole spectrum of possibilities. At one end of this spectrum, trial events can be completely disregarded and queries can be performed on the intersection of the entire CDE sets across the considered trials. At the other end of the spectrum, a user-specified CDE grouping across trials can be used.

The usefulness of either approach is limited by their complexity, so our query generalisation will focus on combining trial events into subsets and using these subsets as the basis for the query generation. This approach addresses the use case mentioned at the beginning of this section, and is both simple and effective in many real-world scenarios.

The generalised cross-trial query is very similar to the basic trial query, except that event sets rather than individual events are mapped on to query elements:

$$
\begin{array}{l}
\underline{\mathit{GeneralisedTrialQuery}} \\
\mathit{trialSet} : \mathbb{P}\ \mathit{TrialDesign} \\
\mathit{queryTerms} : \mathbb{P}\ \mathit{TrialEvent} \nrightarrow \mathbb{P}\ \mathit{QueryElement} \\
\hline
\bigcup(\mathrm{dom}\ \mathit{queryTerms}) = \bigcap\{t : \mathit{trialSet} \bullet t.\mathit{events}\} \\
\forall\ \mathit{events} : \mathrm{dom}\ \mathit{queryTerms} \bullet \\
\qquad \{\mathit{qt} : \mathit{queryTerms}\ \mathit{events} \bullet \mathit{qt}.\mathit{cde}\} = \\
\qquad\qquad \bigcap\{t : \mathit{trialSet} \bullet \bigcup\{e : \mathit{events} \bullet t.\mathit{eventCdeSet}\ e\}\} \\
\forall\ \mathit{events} : \mathrm{dom}\ \mathit{queryTerms} \bullet \forall\ \mathit{qt}1, \mathit{qt}2 : \mathit{queryTerms}\ \mathit{events} \bullet \\
\qquad \mathit{qt}1 \neq \mathit{qt}2 \Rightarrow \mathit{qt}1.\mathit{cde} \neq \mathit{qt}2.\mathit{cde}
\end{array}
\tag{22}
$$

The change from individual events to event sets in the definition of the *queryTerms* component of the query led to a couple of changes to the constraints. The first one relaxes the constraint on the domain of *queryTerms*, specifying it as any set of event sets whose union gives the whole set of overlapping events for the considered trials. The change to the second constraint specifies that the CDE components of the query elements associated with a trial event set are obtained by considering the CDEs associated with all the events in the set.

Notice that the above definition does not require that different event sets involved in the query are disjoint. If this is required, then the domain of the *queryTerms* mapping becomes a partition of the *queryTerms* domain, and the query is specialised to:

$$
\begin{array}{l}
\mathit{PartitionedGeneralisedTrialQuery} \mathrel{\widehat=} [\mathit{GeneralisedTrialQuery} \mid \\
\qquad \forall\ x, y : \mathrm{dom}\ \mathit{queryTerms} \bullet x \cap y \neq \emptyset \Rightarrow x = y]
\end{array}
\tag{23}
$$

If we further constrain the query in (23) so that each element of the partition contains precisely one event, we obtain the *singleton* query below:

$$
\begin{array}{l}
\mathit{SingletonGeneralisedTrialQuery} \mathrel{\widehat=} [\mathit{PartitionedTrialQuery} \mid \\
\qquad \forall\ x : \mathrm{dom}\ \mathit{queryTerms} \bullet \#x = 1].
\end{array}
\tag{24}
$$

This is equivalent to the original trial query in (12).[3]

B. Access Control

The query system needs to be security aware, so that access to confidential information is limited to the rightful users. Clinical trials use a role-based access control (RBAC) approach [18, 19] to constrain data access to users that have certain roles in the trial, and our query system needs to reflect this. Given the set of possible user roles

$$
[\mathit{Role}],
\tag{25}
$$

a RBAC-aware trial design defines the rules that specify the common data elements that each role can query :

$$
\begin{array}{l}
\underline{\mathit{RbacTrialDesign}} \\
\mathit{trial} : \mathit{TrialDesign} \\
\mathit{accessRules} : \mathit{Role} \rightarrow \mathbb{P}\ \mathit{Cde} \\
\hline
\forall\ \mathit{roleCdeSet} : \mathrm{ran}\ \mathit{accessRules} \bullet \\
\qquad \mathit{roleCdeSet} \subseteq \bigcup\{e : \mathit{trial}.\mathit{events} \bullet \mathit{trial}.\mathit{eventCdeSet}\ e\}
\end{array}
\tag{26}
$$

Users that access data corresponding to an RBAC-enabled clinical trial have a view of the clinical trial that is specific to their roles. For the purpose of querying trial data, each of these customised trial views is identical to the view provided by a basic *TrialDesign* instance obtained by filtering out the inaccessible CDEs from the original trial. The following filter maps a (*Role*, *RbacTrialDesign*) pair to its query-equivalent basic trial:

$$
\begin{array}{l}
\mathit{filter} : (\mathit{Role} \times \mathit{RbacTrialDesign}) \rightarrow \mathit{TrialDesign} \\
\hline
\forall\ r : \mathit{Role} \bullet \forall\ t : \mathit{RbacTrialDesign} \bullet \\
\qquad (\mathit{filter}\ (r, t)).\mathit{events} = t.\mathit{trial}.\mathit{events} \\
\forall\ r : \mathit{Role} \bullet \forall\ t : \mathit{RbacTrialDesign} \bullet \forall\ e : (\mathit{filter}\ (r, t)).\mathit{events} \bullet \\
\qquad (\mathit{filter}\ (r, t)).\mathit{eventCdeSet}\ e = \\
\qquad\qquad t.\mathit{trial}.\mathit{eventCdeSet}\ e \cap t.\mathit{accessRules}\ r
\end{array}
\tag{27}
$$

The constraint part of the filter definition describes how the two components of a basic *TrialDesign* (i.e., its event set and its event-CDEs associations) are built from the components of the RBAC-enabled trial. Notice that while the basic trial has the same event set as the RBAC-enabled trial, the filtered event-to-CDEs mappings are obtained by eliminating all CDEs inaccessible to the considered role from the original trial.

Given a set of RBAC-enabled trials and a set of associated user roles[4], a cross-trial query can be obtained by first building the set of query-equivalent basic trials using the *filter* in (27), and then employing a *TrialQuery* for the "filtered" trials. This process is described below:

$$
\begin{array}{l}
\underline{\mathit{RbacTrialQuery}} \\
\mathit{trialSet} : \mathbb{P}\ \mathit{RbacTrialDesign} \\
\mathit{roles} : \mathit{RbacTrialDesign} \nrightarrow \mathit{Role} \\
\mathit{queryTerms} : \mathit{TrialEvent} \nrightarrow \mathbb{P}\ \mathit{QueryElement} \\
\hline
\mathrm{dom}\ \mathit{roles} = \mathit{trialSet} \\
\exists\ q : \mathit{TrialQuery} \bullet q.\mathit{trialSet} = \{t : \mathit{trialSet} \bullet \mathit{filter}(\mathit{roles}\ t, t)\} \land \\
\qquad\qquad q.\mathit{queryTerms} = \mathit{queryTerms}
\end{array}
\tag{28}
$$

[3] A bijection can be defined that maps each *SingletonGeneralisedTrialQuery* to a *TrialQuery*.

[4] Note that the same user can have different roles for each of the trials, hence the use of a set of roles rather than a single role across all clinical trials.

The definition of the *filter* operator in (27) ensures that an *RbacTrialQuery* will include all CDEs accessible to the user with the specified *Roles* and no other CDE.

VIII. Conclusions and Further Work

The consistent use of controlled vocabulary and common data elements [3] creates the potential for data sharing in cancer clinical trials [1]. The query system described in this paper realizes this potential by proving cancer researchers and statisticians with a straightforward means for accessing data across multiple clinical trials.

The basic CancerGrid query system offers complex queries targeted at the CDEs associated with the same trial event for each clinical trial of interest. The extensions presented in the previous section allow users to customize the system so that flexible queries extending across trial event boundaries are possible with minimal configuration effort, and the strict security requirements of clinical trials are thoroughly addressed.

Further work is required to validate the proposed extensions with their user community, and to complete an implementation of the system that provides these extensions. An important characteristic of this implementation will be the ability to dynamically include into the query trial set all trials that the user has the authority to access, with the appropriate queries generated at runtime. A further direction to be investigated is the modelling of queries including elements that refer to CDEs that do not appear in all of the queried trials.

Another cross-trial query approach that CancerGrid is considering involves the use of semantic reasoners [20], and additional work is required to combine this approach with the one described in the current paper. Finally, it is our plan to extend and generalize the cross-project query system to other domains that would benefit from CDE-based data sharing within the respective research fields.

Acknowledgements

This work was supported by the UK Medical Research Council grant G0300648. The authors are grateful to Peter Maccallum, Charlie Crichton, Andrew Tsui and the other members of the CancerGrid team for many insightful discussions during the development of the cross-trial query system.

References

[1] James Brenton, Carlos Caldas, Jim Davies, Steve Harris and Peter Maccallum, "CancerGrid: developing open standards for clinical cancer informatics", Proceedings of the UK e-Science All Hands Meeting 2005, pp. 678-681, http://www.allhands.org.uk/2005/proceedings/.

[2] Steve Harris and Radu Calinescu, "Clinical trials model 1.0", CancerGrid Technical Report MRC/1.4.1.1, June 2006, http://www.cancergrid.org/public/documents/2006/mrc/Report%20MRC-1.4.1.1%20Clinical%20trials%20model%201.0.pdf.

[3] Igor Toujilov and Peter Maccallum, "Common data element management architecture", CancerGrid Technical Report MRC-1.1.2, May 2006, http://www.cancergrid.org/public/documents/2006/mrc/Report%20MRC-1.1.2%20CDE%20management%20architecture.pdf.

[4] Radu Calinescu, "Model-based SOA generation for cancer clinical trials", Best Practices and Methodologies in Service-Oriented Architectures – Proceedings of the 4th OOPSLA International Workshop on SOA and Web Services, Portland, October 2006, pp. 57-71.

[5] Jim Woodcock and Jim Davies, "Using Z. Specification, Refinement and Proof", Prentice Hall, 1996.

[6] Yigal Arens, Chin Y. Chee, Chun-Nan Hsu and Craig A. KnoBlock, "Retrieving and Integrating Data from Multiple Information Sources", Journal of Intelligent and Cooperative Information Systems, vol. 2, no. 2, June 1993, pp. 127-158.

[7] Waiman Cheung and Cheng Hsu, "The Model-Assisted Global Query System for Multiple Databases in Distributed Enterprises", ACM Transactions on Information Systems, vol. 14, no. 4, October 1996, pp. 421–470.

[8] Silvana Castano, Valeria De Antonellis and Sabrina De Capitani di Vimercati, "Global Viewing of Heterogeneous Data Sources", IEEE Transactions on Knowledge and Data Engineering, vol. 13, no. 2, March/April 2001, pp. 277-297.

[9] US National Cancer Institute, The cancer Biomedical Informatics Grid, 2006, https://cabig.nci.nih.gov/.

[10] Rebecca Kush, "Can the Protocol be Standardised?", Clinical Data Interchange Standards Consortium, 2006, http://www.cdisc.org/publications/CDISK_ed.pdf.

[11] Peter A. Covitz, Frank Hartel, Carl Schaefer, Sherri De Corondao, Gilberto Fragoso, Himanso Sahni et al., "caCORE: A common infrastructure for cancer informatics", Bioinformatics, vol. 19, no. 18, pp. 2404-2412, 2003, http://bioinformatics.oxfordjournals.org/cgi/content/abstract/19/18/2404.

[12] Anthony Stell, Richard Sinnott and Oluwafemi Ajayi, "Supporting the Clinical Trial Recruitment Process through the Grid", Proceedings of the UK e-Science All Hands Meeting 2006, pp. 61-68.

[13] Gao Cong, Wenfei Fan, Xibei Jia and Shuai Ma, "PRATA: A System for XML Publishing, Integration and View Maintenance", Proceedings of the UK e-Science All Hands Meeting 2006, pp. 432-435.

[14] Chris Poole, Helen Howard and Janet Dunn, "tAnGo — a randomised phase III trial of gemcitabine in paclitaxel-containing, epirubicin-based, adjuvant chemotherapy for women with early stage breast cancer", 2003, http://www.isdscotland.org/isd/files/tAnGo_protocol_version_2.0_July_2003.pdf.

[15] Elaine Beller, Val Gebski and Anthony Keech, "Randomisation in clinical trials", Medical Journal of Australia, vol. 177, Nov. 2002, pp. 565-567, http://www.mja.com.au/public/issues/177_10_181102/bel10697_fm.pdf.

[16] Earl H M, Poole C J, Dunn J A, Hiller L, Bathers S, Spooner D et al, "NEAT - National Epirubicin Adjuvant Trial, a multi-centre phase III randomised trial of Epirubicin x 4 and classical CMFx4 [ECMF] versus CMFx6", Proceedings of the American Society of Clinical Oncology **21** (2), 2002, pp. 1081-0641.

[17] Bill Evjen, Scott Hanselman, Farhan Muhammad, Srinivasa Sivakumar and Devin Rader, Professional ASP.NET 2.0, Wiley Publishing, 2006.

[18] D.F.Ferraiolo, D.R.Kuhn and R.Chandramouli, Role-Based Access Control, Computer Security Series, Artech House, 2003.

[19] David F. Ferraiolo, Ravi Sandhu, Serban Gavrila, D. Richard Kuhn and Ramaswamy Chandramouli, "Proposed NIST standard for role-based access control", ACM Transactions on Information and System Security, 4(3), pp. 224–274, August 2001, http://csrc.nist.gov/rbac/rbacSTD-ACM.pdf.

[20] V. Haarslev and R. Moller, "Racer system description", Proceedings of the First International Joint Conference on Automated Reasoning, Lecture Notes in Computer Science, vol. 2083, Springer-Verlag, 2001, pp. 701-706.

User Interfaces for an In-store Sales Process Supporting System

Rainer Blum[1], Sascha Häberling[1], Karim Khakzar[1], Steve Westerman[2]

[1]Fulda University of Applied Sciences,
Dep. of Applied Computer Science
Marquardstr. 35, D-36039 Fulda, Germany,
{rainer.blum, sascha.haeberling, karim.khakzar}
@informatik.hs-fulda.de

[2]University of Leeds,
Institute of Psychological Sciences,
Leeds LS2 9JT, UK
s.j.westerman@leeds.ac.uk

This paper considers the application of computer-based technologies in retail settings and describes the development of a system designed to support the in-store interaction between customer and sales person. The particular application context is that of a made-to-measure shirt shop. A sustained program of research has led to the development of a system that facilitates the selection of shirt components (e.g., fabrics, collars, cuffs) and allows customers to see a virtual representation of a shirt prior to purchase. An iterative and participatory design process has been adopted, and many interface alternatives considered. Results of this work from a usability point of view are presented and implications considered. Advantages for the customers, sales personnel, and shop owners can be identified. However, integration of 'usable' computer technology in this complex 'real world' setting needs to be improved and further issues remain to be resolved.

I. INTRODUCTION

Information technology is pervasive in today's work and home environments. It is difficult to imagine daily life without computers. The retail trade has been at the forefront of much of this development, with e-commerce becoming an increasingly important retail method. According to TNS Infratest, analysts expect West-European e-commerce turnover to rise up to approximately 2.4 billion Euros in 2007, which is three times more than generated in 2004 [1].

There are many desirable features of e-commerce. For example, it provides consumers with access to a wide variety of product alternatives and makes economies of scale possible. However, e-commerce cannot service all consumer needs. For example, social aspects of the shopping experience are not easily substituted within such systems. Related to this, e-commerce does not provide consumers with the immediate support of a sales assistant, nor does it provide access to real world products for appraisal. These latter factors may be particularly important to consumers when selecting certain types of products, e.g., those for which physical characteristics are key (such as fabrics), expensive products, and/or non-standard or customizable products. Moreover, consumers vary in their familiarity with and access to computer technology. Some groups in society will be less likely to use e-commerce and more likely to prefer, or to rely on, 'real world' shopping.

Given these potential advantages of 'real world' shopping, it is valuable to consider means by which the complex real world sales process can be supported by virtual technology (see e.g. [2]). In this context, computer-based support might take several forms, including decision support [3], [4] and product illustration [5]. However, as with other settings in which there is a similar interplay between user, advisor, real objects/environment, and virtual objects/environment (e.g., augmented reality systems in museums [6]), delivery of a usable system is not a trivial task.

In this paper we present an overview of a sustained program of research that has focused on the development of an in-store sales support system. The particular sales context is made-to-measure shirts. The goal of this system is to support the interaction between sales person and customer – not to replace the sales assistant. It should facilitate the selection and comparison of product alternatives. When shopping for a made-to-measure shirt the customer would typically compose a desired shirt from a large array of fabrics, cuff designs, etc. To support this, examples of fabrics, product components (e.g., pockets or collars), and some made-up shirts are likely to be available to the customer. However, customers are usually not able to see a shirt made to their full specifications prior to purchase and production. Especially new customers often have difficulties imagining what the shirt would finally look like and thus deciding which options to choose.

Computer-based technology has the potential to support the selection process, improving access to the large database of design alternatives. In addition, virtual representations of product alternatives would obviously be helpful, and this has been one of the primary goals of this research work. This would enable the customer to get a clearer understanding of what the selected combination of shirt design features would look like when the shirt is completed.

Usability is essential for the success of any interactive system and, thus, is another major goal of this work. Our results in this field are reported hereafter.

Work started with the ShopLab project, 'The Network for Test and Design of Hybrid Shop Environments based on multi-modal Interface Technology' [7] in November 2001 and was seamlessly continued with IntExMa, the 'Interactive Expert System for Made-To-Measure Clothing' [8] in Octo-

This research was part funded by the European Commission under the Information Society Technologies (IST) priority, project title 'ShopLab', contract no. IST-2000-28143, and the German Federal Ministry of Education and Research (BMBF) in the program 'Angewandte Forschung an Fachhochschulen im Verbund mit der Wirtschaft' (FH³), project title 'IntExMa', contract no. 17 06A 04.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 391–396.

ber 2004 which is still underway. ShopLab aimed to develop hybrid multimodal systems for traditional city-centre shops that would be applicable within a wide variety of shops and different cultural contexts throughout Europe. It was important that the solutions were customisable to enable support and enhancement of the distinguishing characteristics of individual shops. In contrast, IntExMa focuses on the improvement and optimization of the previously mentioned solution for a made-to-measure shop, which arose during ShopLab.

In the following sections of this paper, first our methodical approach is described followed by a description of the basic system concept. Subsequently, a historical account of the developmental process is provided, broken down into three sections, each including a system description, details of user testing, results of testing, design decisions, and implications for the following design iteration. Finally, conclusions are drawn and future directions considered.

II. Methodological framework

We implemented a User-Centred Design (UCD) process that incorporated the following steps: i) analysis of context of use; ii) requirements specification; iii) production of design solutions; iv) evaluation of design results with (both) end-user groups; and, v) feedback into the next design cycle [9]. In total, three iterations were completed, during which different interface alternatives were considered, implemented as rapid prototypes and tested in the project laboratories with the most promising finding their way into the superordinated system. Test participants were always matched to the demographic profile of shirt shop customers and sales staff. Tests with physically handicapped people that have to use a wheelchair were also carried out, as this group of consumers has particular difficulties in many traditional shopping environments.

During regular evaluation phases the complete system was subject of extended usability and acceptance tests, conducted with real customers and sales staff in one of the shops of the made-to-measure tailor Campe & Ohff in Berlin or Hamburg. In this way, usability data were gathered in the same environment as the final product would be used in.

The tests always included quantitative and qualitative parts: Formal task-based tests were carried out along with informal observations and discussions with users. The formal tests consisted of a series of tasks designed to represent key components of the system functionality followed by a short questionnaire. In all cases the objective of the tests was to gain detailed feedback about the usability of the system for each of the main system features and functions.

III. Overall system concept

The context of use analysis was designed to provide a comprehensive understanding of the characteristics of the users, tasks, and the organisational and physical environment. From this a precise specification of the initial system requirements was generated, that took into account a usability perspective. Two user groups were differentiated: customers and sales staff. For the latter, basic IT experience can be expected, and some training on the developed system will be possible. When considering the customers a wide range of characteristic values (age, abilities, prior knowledge, physical handicaps) had to be assumed. This required taking into account the lowest common denominator of user skills for the system design.

General patterns of activity in sales/shopping behaviour were identified and analysed: Individuals engage in a process of collaborative consultation, with sales staff acting as experts to identify customers' requirements, and to guide them through the process of selecting an appropriate product. A large part of this process can be described as a complex decision-making sequence. An initial break down of the product range into groups of items of interest is followed by a detailed consideration of identified suitable options. These are considered in greater detail, as the customer gradually focuses on a single item of interest.

Designed as an in-store, stand-alone installation, the system incorporates all the hardware and software necessary for its operation. From the user perspective, a large-scale, vertically mounted display screen (minimum 40 inches screen size) forms the centre. Its purpose is to display the virtual 'try-on', which represents one of the central ideas behind the system: It allows the customer to see themselves, or at least a virtual representation of their body, dressed in the desired made-to-measure shirt, also as a virtual representation. This set-up follows the metaphor of a 'virtual mirror'.

During the course of this research, two different approaches were considered to accomplish the desired effect. The first utilised augmented reality technologies, the second virtual reality (see next sections). To enable the customer and the sales staff to interact with the virtual scene, e.g. see the shirt from different angles or from behind or zoom into details, a range of special interaction techniques, devices and user interface concepts were researched.

The idea of using a 'virtual mirror' for the sales of made-to-measure clothing and clothing in general is not brand-new. Several projects and companies have already worked on its implementation or the solution of related sub problems. The approaches taken differ a lot and have varying levels of complexity, see e.g. [10] for a very sophisticated solution and [11] for a less complex web-based solution. The innovative aspects of the research reported here results from the system's direct integration into the sales process and the consideration of the added-value to small and middle sized retailers and their customers. Thus, innovative, technically high-grade, but affordable solutions were the focus of the project.

The second functional area of the system, closely related to the virtual 'try-on', is the product selection element (choosing from the range of product components offered by the shop) with a large database of selectable design alternatives, customer data management (e.g. name and address, body measurements, delivery modalities, product favourites and sales history) and electronic sales features (e.g. shopping cart).

The functional domains of 3D interaction and business process support were not considered as isolated with well-separated interfaces. Instead, the aspiration is to integrate both in an effective way.

The overall goal of the installation is to give the customer a clearer impression of what their shirt will look like, before it is actually manufactured. The system can be used by the customer to choose from all the components offered by the shop to compile his or her shirt of choice. Sales personnel also interact with the system to support the customer while he or she is in the shop. The system is not intended to replace the sales personnel but provides a complementary tool to support and enrich the shopping process.

IV. AUGMENTED REALITY, STATIONARY TOUCHSCREEN

As mentioned, several different versions of the system have been researched in the course of the two projects, and these are described in this and in the following two sections.

A. *System description*

The first system version was based on Augmented Reality (AR). This research field deals with the combination of real world and computer-generated data. At present, in most cases, live video imagery is captured, digitally processed and 'augmented' by the addition of computer-generated graphics.

For the system at hand the existing software ARToolkit [12] was used, which is able to recognize pre-defined black-and-white symbols printed on a piece of card and calculate its position and angles in space relatively to the camera.

The required camera was mounted in the system installation in such a way that it captured the user standing in front of the large display in a 'mirror-like' fashion. The live video is augmented with a three-dimensional representation of the desired shirt, the location and nature of which is determined by the location and nature of the symbol card. Thus, customers can simply use the cardboard to interact with the virtual scene. They can control the position and angle of the displayed shirt and coverage of their mirrored image by tilting and moving the symbol in front of their body (see Fig. 1).

All the other functionality of the system can be accessed via a separate touch screen unit with a graphical user interface (GUI). It was primarily designed for usage by the customers, accompanied, if required, by the sales assistant. Its main component is the electronic product catalogue, that can be browsed and filtered and allows the configuration of the individual shirt. Every change in the configuration, e.g. choosing a different cloth, results in direct change to the virtual shirt. The shirt is permanently presented on the virtual 'try-on' display, assuming the symbol card is visible for the camera.

Fig. 1: Augmented Reality prototype with touch screen (left), large display (right) with video camera on right edge and test person with symbol card

Besides, the customers can access a personal user space with personal contact details and order the configured shirt.

B. *Usability test results*

Although most participants were able to use the symbol cards after the tester had provided a short demonstration, many found it difficult to match the shirt to their body acceptably. The system was badly affected by lighting conditions and, therefore, performance altered according to time of day. The symbol cards only worked effectively when held close to the body at a slight angle and this was not at all intuitive. Many participants held the symbol cards in the wrong position, therefore covering the critical area for recognition by the camera.

Although participants found the system to be fun, they did not find it useful for the selection and 'try-on' of shirts. This was also partly due to the limited range of choices available within this version of the system, together with limited visualization quality. Many of these points are general problems that occur in AR systems. Solving them would have required a technical effort beyond the resources of the project. These points showed that the augmented reality approach might not be a good solution for the system and that other technologies should be researched in the context of the application.

When considering the performance of the touch screen GUI, participants generally found it easy to compose a shirt. Despite this fact and positive opinions about the touch screen itself, users complained about the way in which information is organized in the system. They stated that the electronic product catalogue differs too much from the way the physical shop is organized.

In summary, it appeared that the system was fun but not useful to the customer. Single components like the touch screen showed potential, but needed extended functionality in order to make the overall system concept more successful. The system also needed to be integrated more effectively with the real shop and to provide better support for the process of collaborative consultation between customer and sales staff.

Due to the identified drawbacks of the AR-approach, with required technological improvements appearing not viable in the scope of the project, it was decided to switch to a Virtual Reality (VR) solution.

V. VIRTUAL REALITY, STATIONARY TOUCH-SCREEN, KNOB AND BUTTONS

The main development tasks for the VR-based virtual 'try-on' were to implement a virtual representation for the customer, to drastically advance the appearance of the virtual shirts towards high realism and to find appropriate ways to interact with the virtual scene. Concerning the user interfaces, the identified usability issues needed to be addressed.

A. System description

This system (see Fig. 2) provides three-dimensional figurines (avatars) as virtual counterparts for the customers. The latter could choose their personal favourite from a range of predefined male figures.

These were identified as sufficient to provide adequate representations of the large majority of users including a sitting avatar for wheelchair users. The avatars remain in a static posture and are not able to move their limbs.

The selected avatar is always shown wearing the currently configured shirt, which is prepared to fit almost perfectly. The virtual environment is completed with a background image with realistic lighting.

To interact with the avatar and the virtual scene a special, commercially available interaction hardware device [13] is used that incorporates a rotating knob and a range of buttons. Customers can turn the virtual figure around its vertical axis and invoke predefined view points in the sense of zooming into details, e.g. the cuffs or the collar, using these controls.

The design of the touch screen solution was guided by the scenario of the system being used by a customer accompanied by a sales assistant. This allowed the inclusion of more complex features, although intuitive interface designs remained an important goal.

An electronic product catalogue was integrated with a bar-code scanner in the form of a wired pen. Markers were added to the real product components in the shop, so that customers and sales staff can select the desired shirt features from this product range. As before, changing the shirt configuration directly results in the relevant changes to the virtual shirt. The personal user space was extended and now offers saving favourite shirt components and complete shirt configurations.

Fig. 2: Virtual Reality prototype with touch screen (left), ShuttlePRO device (attached right hand of the touch screen) and 3D scene display (right)

B. Usability test results

The use of avatars worked well for most users but some complained about the look of the available avatars and that they could not find a model that they felt was an appropriate representation of their body. The four avatars of choice were modelled to represent a cross-section of body shapes, but they could not represent the body shape of every customer. The user tests showed that, overall, the shop customers found the avatar image to be neither realistic nor unrealistic in terms of providing an impression of body-shape. They also provided neutral ratings for overall image quality.

The input device used for the 3D interaction worked very well. It showed great potential for a minimal, reliable, but effective hybrid system.

The bar-code scanner facility was moderately successful. Problems were caused by its unreliability with certain bar codes. The nature of the stick format required that the users to wipe it over the bar code quite precisely. Also the cable was criticized as being clumsy. These negative points would be easy to eliminate by using another kind of device. Overall, the bar-code scanner idea performed well for its purpose, enabling the composition of the virtual shirt and integrating the system with the real range of products in the shop.

The users' main criticism about the system was that the touch screen interface was too complex and forced them to follow unnecessary steps to access useful information. This problem arose mainly from the comprehensive component catalogue, hindering an intuitive usage without a significant, tester supported learning phase.

Usability tests performed with the wheelchair users showed that the system was advantageous for them. They were able to use the same functionality and could interact with the virtual scene to the same extent as other users.

Overall, from a usability perspective, the results were encouraging and suggested that, with feasible modifications, the shirt shop prototype would have the potential to work effectively within real-world shopping contexts. Indeed, it was successfully used with real customers and was instrumental in supporting their purchase decisions.

VI. Virtual Reality, Custom Knob and Buttons, PDA

With the changeover to the new project 'IntExMa', a more realistic virtual representation of the clothing and the customers became a major focus, as these remained prominent points of criticism preventing an effective virtual 'try-on'. Usability persisted as a second main focus, taking the insights of the previously conducted work as starting point.

A. System description

Concerning the virtual 'try-on', the VR-approach was kept. Again, customers can choose from a range of male avatars. These were remodelled from scratch and with greater details for a more natural appearance. Instead of dressing them with predefined clothing, the shirt representations are generated dynamically for the virtual figures. Also, the lighting was made much more sophisticated resulting in a greater degree of realism of the drape of the shirts and their general appearance.

The view of the system progressed from one of a system whose complete functional spectrum can be used by customers (though accompanied by the sales staff), to one that would only offer selected functions to them. This aspect resulted from interim tests, in which only the sales assistant interacted with the system, as a means of decreasing the complexity for the customer and maximising provision of service.

However, the great majority of the customers involved in the tests wanted to have access to some of the systems functions independently. As a consequence, two different user interfaces are offered, a PDA for the sales staff and an interaction device consisting of only two buttons and a rotating knob for the customers.

Via the PDA, the salesperson has access to all the functionality of the system: electronic product catalogue with capacious ordering functions, customer data and 3D interaction. Concerning the latter, the PDA works like a remote control that transfers changes in the actual shirt configuration directly to the virtual 'try-on', provides avatar selection and allows turning the avatar and zooming in to details of the shirt. The popular bar-code scanner is now integrated in the PDA.

The knob and buttons device (see Fig. 3 and 4) was custom-designed, as no commercially available device could be found, that provides the desired form factor, simplicity and robustness of construction. The required, associated GUI is overlaid on top of the 3D product visualization.

With the 'confirmation' button the user invokes the GUI respectively the product catalogue, switches to different levels of its hierarchical structure, and chooses the components for the shirt, again directly illustrated in the 'try-on'. The product detail currently configured is at the same time automatically zoomed in the 3D scene. The second button, 'abort', provides going back in the navigation path and, having reached the top level, fading out the GUI. Browsing the different entries inside a menu level is achieved with the rotary knob. With its help, the avatar can also be rotated, presumed that the GUI is not visible. Both interfaces, PDA and knob and buttons interface operate on the same data and can be used concurrently.

B. *Usability test results*

Task-based user tests, embedded into a simulated customer-salesperson dialog, revealed only minor usability problems and high functional usefulness for the PDA, both for business process functionality and 3D scene interaction. Admittedly, free text entry was an issue, due to the small software keyboard or in the case of handwriting recognition a sometimes unsatisfactory recognition rate. Most people tested stated that more practice would help a lot, but nevertheless, further research into these aspects is considered worthwhile.

The mobility of the PDA was generally appreciated, but, on the other hand, concerns about the general form factor of the device could be identified on the part of the sales staff: Some stated that customers needed a definite point of reference in the shop where the order process can be completed in a confidential atmosphere. Nevertheless, the customers liked the idea

Fig. 3: PDA (3D interaction mask) and knob and buttons device

of being supported by the sales person in this way during the purchase process, though some said they would like to get more insight into the actions performed on the PDA. Therefore, an 'on demand' integration with stationary IT equipment should be considered in future versions. Apart from that, some users raised concerns that the sales person could be concentrating too much on the PDA device and thus not paying sufficient attention to the customer.

The knob and buttons device with its connected GUI was very successful. Users had only minor criticisms. The customers could perform all the given tasks quickly without too much effort in both domains - component selection from large product range and 3D scene interaction.

A point that should be kept in mind concerns the way people feel about their virtual counterpart when confronted with it. Shop owners raised concerns that the appearance of the

Fig. 4: System prototype at the CeBIT 2006 fair, Hanover, Germany; knob and buttons device (in the middle) and 3D product visualization with superimposed GUI showing the top menu level (right)

avatar, with possible quality shortcomings, could be more attention grabbing than the shirt itself. Quite often customers made remarks about specific characteristics of the avatar, e.g. its hair style or body posture and about what could be improved in its appearance instead of concentrating on the product, the individual made-to-measure shirt.

VII. CONCLUSIONS AND FURTHER STEPS

The ShopLab project and its successor IntExMa have focused on the usability of a system designed to support the consultation and sales process. Simple user interfaces have been developed providing intuitive and easy interaction with large information spaces and virtual reality scenes. In the course of this research and development a number of different approaches and concepts have been explored. These have led, in logical steps, to the current solution.

The general concept of bringing these types of technology into traditional inner-city shops in a hybrid manner was appreciated by customers as it enriches their shopping experience. Users liked the combination of the systems' virtual information with the real products on the shelves, praised its publicity value and regarded the ability to visualize information as useful. Interaction with the virtual scene via the offered interfaces proved to be easy for them, although the quality of the shirt and avatar representation remained subject to some criticism. The virtual 'try-on' concept was very popular with users with mobility problems as it bypasses the need to use changing rooms. Trying on clothes in normal shops is very difficult for these users.

The approach taken in this research was to regard both functional domains covered by the developed user interfaces, 3D interaction and business process support, to be interdependent and, on the whole, the goal of effective integration was successfully achieved. However, competing requirements had to be considered and some usability issues require further work, e.g. whether rich PDA-functionality has a negative impact on the concentration and attention of the sales person.

The knob and buttons interface concept proved quite popular, but confirmatory evidence is required when it is used in conjunction with more complex functional structures.

The customers' attitudes towards the characteristics of their digital representation have proved to be quite controversial and also deserve further study. The identification of the user with their virtual counterpart might be improved if, in addition to selecting an avatar from a given range, the user was able to customize the avatar further, e.g. changing hair and skin colour. An avatar that was able to move its limbs may also result in more positive reactions. On the other hand, it is not clear if a high degree of personal identification is necessary or even desirable. After all, the product should be the main focus of attention while the other virtual elements fulfil a kind of framing function.

Our vision and principle for further research is to ubiquitously integrate information technology in the retail business and other everyday application settings and in that way to support the idea of participation for all in the Information Society. Our approach is to utilize the benefits of virtual reality, e.g. turning mere data into graspable information, in common usage contexts. Our results suggest that this must take place via a natural integration without technically complex 3D interaction devices. In the shopping context such solutions can allow progression beyond the constraints imposed by reality, adding a rich sensory quality, whilst maintaining a high level of utility.

One of our next steps will be to tackle the problem of effective production of the required product content (clothing patterns, product photos etc.) while improving the quality of the 3D visualization. Furthermore, we want to explore the feelings people encounter when faced with virtual counterparts that resemble them. In this context, we also need to clarify how the system can be provided with the necessary customization information. Automatic mechanisms will result in a technically more sophisticated system, which conflicts with our basic simplistic approach. Engaging the sales staff or the customers for this task is an additional activity that may conflict with the natural sales process.

We also plan to put a stronger focus on handicapped and elderly people, as our concept proved to be advantageous for them. This will be accomplished in combination with our wish to transfer our findings to other application areas e.g. special support systems for the elderly in thematic areas such as improved autonomy and lifelong learning.

REFERENCES

[1] TNS Infratest, Monitoring Informationswirtschaft, 2005, http://www.tns-infratest.com/06_BI/bmwa/Faktenbericht_8/pdf/FB8_Vollversion_de.pdf, Retrieved Oct. 5, 2006.

[2] T. Rodden, Y. Rogers, J. Halloran and I. Taylor, Designing novel interactional workspaces to support face to face consultations, Proceedings of CHI, ACM, 2003.

[3] G. Haubl and V. Trifts, Consumer decision making in online shopping environments: The effects of interactive decision aids, Marketing Science, 19, 4-21, 2000.

[4] S.J. Westerman, G.C. Tuck, S.A. Booth and K. Khakzar, Computer decision support systems: Internet versus in-store application, Computers in Human Behavior, in press.

[5] Eon Reality, Inc., Case Study Suzuki, 2002, http://www.eonreality.com/Casestudies/suzuki/index.pdf, Retrieved Oct. 5, 2006.

[6] D. Hywel, E. Roberts and J. Everitt, A study of information and communications technologies in museums with particular reference to Wales, Program: electronic library and information systems, vol. 33, no. 4, 1999.

[7] ShopLab, research project sponsored by the European Commission, 2004, http://www.shoplab.info, Retrieved Oct. 5, 2006.

[8] IntExMa, research project sponsored by the German Federal Ministry of Education and Research, 2006, http://www.intexma.info, Retrieved Oct. 5, 2006.

[9] ISO, International Organization for Standardization, ISO 13407, Human-Centred Design Processes for Interactive Systems, Geneva, 1999.

[10] Virtual Try-On, research project sponsored by the German Federal Ministry of Education and Research, http://www.igd.fhg.de/igd-a3/presentations/englisch/VirtualTryOn/index.htm, Retrieved Oct. 9, 2006.

[11] My Virtual Model Inc., http://www.mvm.com/, Retrieved Oct. 9, 2006.

[12] M. Billinghurst and H. Kato, Collaborative Mixed Reality, Proceedings of International Symposium on Mixed Reality (ISMR '99), Mixed Reality--Merging Real and Virtual Worlds, 261-284, 1999.

[13] Contour Design, Inc., ShuttlePRO V.2 and ShuttleXpress Multimedia Controller, http://www.contourdesign.com/, Retrieved Oct. 10, 2006.

Histochemical and mechanomyographical evaluation of the adaptive potential of human biceps femoris muscle

Raja Dahmane, MD, PhD
College of Health Studies, University of Ljubljana, Slovenia

Abstract - **The goal of this study was to estimate the ability of biceps femoris (BF) muscle, a hamstring muscle crucial for biarticulate movement, to adapt to changed functional demands. For this purpose and due to ethical reasons, in a group of healthy sedentary men and of 15 sprinters, a non-invasive mechanomyography (MMG) method was used to measure the muscle twitch contraction times (Tc). These correlate with the proportions of slow and fast fibers in the muscle. To further elucidate the data obtained by MMG method and to obtain reference data for the muscle, the fiber type proportions in autoptic samples of BF in sedentary young men were determined according to histochemical reaction for myofibrillar adenosine triphosphatase (mATPase). With MMG we indirectly demonstrated that biceps femoris muscle has a strong potential to transform into a faster contracting muscle after sprint training, since the average Tc in sprinters was much lower (19.5 ± 2.3 ms) than in the sedentary group (30.25 ± 3.5 ms). The results of the histochemical analysis of BF muscle also imply a high adapting potential of this muscle. Beside type 1, 2a and 2× (2b) fibers a relatively high proportion of intermediate type 2c fibers (5.7% ± 0.7), which co-expressed MyHC-1 and -2a, was found. Therefore, type 2c might represent a potential pool of fibers, capable of transformation either to slow type 1 or to fast type 2a in order to tune the functional response of BF muscle according to the actual functional demands of the organism.**

I. Introduction

The functional characteristics of skeletal muscles depend on the muscle fiber properties. The fiber type proportions differ among different muscles and depend not only on their functional role, but on other factors, such as innervations and the pattern of mechanical stimuli as well (1). In response to different stimuli, the muscle fibers are able to adapt their properties with a change in the MyHC isoform expression, the process known as fiber type transition. Such a dynamic nature of muscles has been known as muscle plasticity (2).

There are very few reports on the histochemical structure and on the effect of short-term high-intensity training on the contractile properties of the biceps femoris muscle (BF). As one of the posterior femoral or hamstrings muscles (semitendinosus, semimembranosus and biceps femoris), BF is frequently injured during sports activities, sports injuries most commonly appear in the running section of athletics, especially in the first 10-20 m of a sprint (3). BF crosses the posterior aspect of both the hip and the knee joints. The flexion of the knee and the stabilizing effect are very important function of this muscle. The extension of the hip joint, when the thigh is the moving part, is another important function, particularly when the trunk is bent forwards and is to be raised to the erect position.

Since it has been demonstrated that a strong correlation exists between the whole-muscle twitch contraction time, measured by tensiomyography (TMG) and the histochemically determined percentage of slow twitch or type 1 muscle fibers, i.e. the higher type 1 proportion the higher the contraction time, the TMG measurement method was proposed as a non-invasive alternative to an invasive histochemical analysis (4,5). TMG can be classified as mechanomyography (MMG) which is investigating the mechanisms of muscle contractile properties with a particular reference to the mechanical nature of the measurements. This method is based on the assumption that radial belly displacement detected by an optomagnetic sensor is proportional to the muscle force. The procedure has been evaluated in healthy young subjects (4, 5) and by subjects after above knee amputation (6). With the proposed method the response of a single muscle within a given muscle group can be measured. The aim of this study was to estimate the extent of fiber type transitions in BF muscle after at least five years of specific sprint training. Since the histochemical classification of fiber types could not be performed on byoptic samples due to ethical reasons, the non-invasive mechanomyography method was used to determine the contractile properties (contraction time) of BF muscle, as a reflection of the ratio between the slow and fast fibers in group of sedentary and of sprint trained young men. To further elucidate the data obtained by MMG method and to obtain a basic reference data for the muscle, the fiber type proportions in autoptic samples of BF of sedentary young men were determined histochemically.

II. Material And Methods

Subjects

A non-invasive MMG method was used for the measurement of skeletal muscle contraction, expressed as contraction time (Tc), in a group of 15 healthy sedentary male subjects (17-40 years old), who volunteered for the investigation and in a group of 15 male sprinters (23.2 ± 3.1 years old, height 1.79 ± 0.05 m and body mass 77.6 ± 7.7 kg), who were recruited for this study.

T. Sobh (ed.), *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, 397–401.

Muscle samples

Biceps femoris muscle samples were taken at autopsy from a group of 15 male subjects, aged between 17 - 40 years, who had died suddenly (suicide, traffic accident). The autopsies were performed within 5 to 24 hours after death. The local ethical committee approved the muscle sampling. The background of the autopsied subjects is known from the legal medicine documentation. The post-mortem examination revealed no significant pathological changes other than those related to the immediate cause of death.

Muscle blocks (approximately 1 cm3) were frozen rapidly in liquid nitrogen cooled to –196 °C. Serial sections (10 µm) were cut in a cryostat at –20°C and processed for the histochemical demonstration of mATPase activity.

To compare the data obtained from BF muscle, the fiber type proportions assessed by the histochemical reaction for mATPase and the data obtained by mechanomyography in other lower and upper limb muscles (biceps brachii, triceps brachii, brachioradialis, flexor digitorum superficialis, extensor digitorum, biceps femoris, tibialis anterior, gastrocnemius – caput mediale and soleus), analyzed in our parallel study, are presented in this paper as well.

MATPase histochemistry

Type 1, 2a, 2x (2b) and 2c fibers were determined according to the activity of mATPase reaction using the calcium histochemical method with alkaline preincubation at pH 9.4 and acid preincubation at pH 4.6 and 4.3. The fast fibers display a high mATPase activity under alkaline conditions and low activity under acid conditions, whereas slow fibers exhibit the inverse. Type 2a fibers stain deeply after alkaline preincubation (pH 9.4) and lightly after preincubations at pH 4.6 and pH 4.3, type 2x (2b) stain deeply after alkaline preincubation, moderately at pH 4.6 and lightly after preincubation at pH 4.3. Type 2c fibers are stable to various degrees throughout the pH range from 4.3 to 9.4 (7).

In each muscle sample an area was selected at random and was photographed by Opton photomicroscope with a constant magnification of 116×, so as to include at least 2 fascicles and a total of at least 100 fibers. The contours of the fibers within the selected fascicles were digitized with the aid of a Cherry graphic tablet coupled to an IBM PC/AT - compatible computer. The percentage of muscle fiber types was determined using the computer - aided method of reference (8).

Mechanomyography (MMG)

The contraction times (Tc), defined as the time between 10% and 90% of the maximum value of the muscle response, of the right BF muscles were measured. The measured subject was laying on his front on the measuring bed. The sensor location was determined anatomically according to reference (9). Maximal amplitude/response was used as an additional criterion for an optimal sensor position. The BF muscle was measured at the midpoint of the line between the fibula head and the ischial tuberosity. The muscle was stimulated with a single twitch stimulus using two self-adhesive electrodes placed symmetrically to the sensor. The anode was placed distally and the cathode proximally, 20-50 mm from the measuring point. The bipolar electrical stimulation used, consisted of a single DC pulse of 1 ms duration and supramaximal intensity (the current was increased gradually from the threshold to the maximal displacement amplitude/response of the muscle).

For the measurements an inductive sensor incorporating a spring of 0.17 N/mm was used. It provides an initial pressure of approximately 1.5 × 10-2 N/mm2 on a tip area of 113 mm2. The measured leg was fastened to the frame with one or two bands to achieve the isometric condition during the measurement.

The measured muscle responses were stored and analyzed with a software package (TMG-BMC) running on PC. The MMG signals were analyzed to determine the contraction time of the muscle response.

Statistical analysis

All three experimental groups of subjects were homogenous concerning the age, sex, and health condition. The data obtained with the histochemical methods were statistically analyzed with the SYSTAT (1991) packet. The results were expressed as mean values and were given plus or minus one standard error. They were compared by analysis of variance and Bonferroni test. The data obtained with the MMG method were statistically analyzed with the MATLAB STATISTICAL TOOLBOX. The results were expressed as mean values and were given plus or minus one standard deviation. The relationship between the maximum running velocity data and the contraction time of muscle belly was tested using the Pearson's correlation coefficient. $P<0.05$ was taken as the limit of significance in all statistical tests.

For the test retest reliability, we used the protocol suggested by reference (10)

Results

Fiber types and MyHC isoform expression in biceps femoris muscle of sedentary men.

Using the histochemical reaction for mATPase, type 1, 2a, 2x (2b) and 2c fibers were demonstrated in BF muscles (Fig. 1 a-c). The mean percentages ± SEM of each fiber type are illustrated in Table 1, in which the fiber type data for different limb muscles (biceps brachii, triceps brachii, brachioradialis, flexor digitorum superficialis, extensor digitorum, biceps femoris, tibialis anterior, gastrocnemius – caput mediale and soleus) of the same subjects, analyzed in our parallel research, were added for the purpose of comparison. BF muscle had similar proportions of type 1 (49.0 ± 1.6 %) and type 2 (51.0± 1.7 %) fibers. The proportion of type 2a fibers (25.2 ± 1.3 %) was slightly higher than that of type 2x (20.7 ± 1.4 %). In comparison with other muscles, BF had a substantial percentage of the intermediate type 2c fibers (5.7 ± 0.7 %). In the analyzed samples there were no obvious signs of fiber type grouping or atrophy.

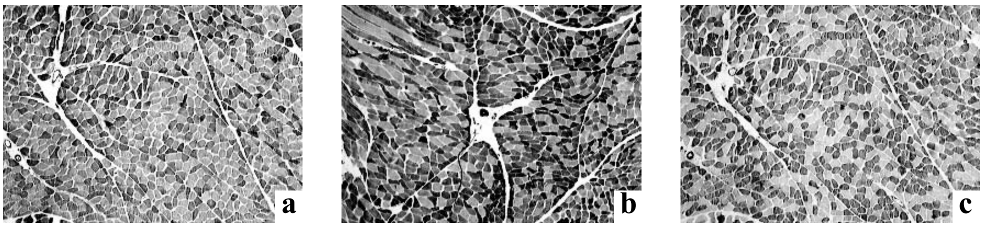


Fig.1: Fiber types (1, 2a, 2x and 2c) in human biceps femoris muscle demonstrated according to the reaction for mATPase at different preincubations (a. 4.3, b. 4.6, c. 9.4)

Table 1: Proportions (mean percentage ± SEM) of type 1, 2a, 2x and 2c fibers and contraction times (mean values ± SD) assessed by mechanomyography of the human limb muscles in the group of sedentary young men (n=15).

muscle	muscle	fiber	type		Contraction times/ ms ±SD
	1	2a	2x (2b)	2c	
Biceps brachii	46.4±2.1	33.4±2.2	16.6±1.9	3.2±0.7	28.87 ± 4
Brachioradialis	47.1±2.2	28.8±2.5	19.5±2.5	2.9±1.7	23.70 ± 5
Triceps brachii	38.0±1.6	39.9±2.0	17.9±1.8	3.9±1.0	22.56 ± 6
Flexor digitorum superficialis	45.3±1.9	41.8±2.4	10.7±1.7	2.0±0.5	25.40 ± 6
Extensor digitorum	47.5±1.7	44.5±1.8	6.5 ±1.7	1.3±0.6	25.26 ± 5
Biceps femoris	49.0±1.6	25.2±1.3	20.7±1.4	5.7±0.7	30.25 ± 3.5
Gastrocnemius	69.7±2.0	24.1±1.7	4.3±0.7	1.8±0.6	44.75 ± 4
Soleus	90.6±2.3	8.7±2.0	0.3±0.2	0.2±0.1	46.52 ± 4
Tibialis anterior	72.9±2.2	20.5±1.6	4.1±1.2	2.0±0.5	32.83 ± 4.5

Contraction times of biceps femoris muscle assessed by mechanomyography in sedentary young men and sprinters

In sedentary men the contraction time (Tc) of the belly twitch response of BF was higher (30.25 ± 3.5 ms) than those of the upper limb muscles and lower than Tc of the lower limb muscles, analyzed in our parallel study. These results are in accordance to the type 1 fiber proportions in the analyzed muscles (Table 1). In Table 2 BF contraction times of the 15 sprinters are shown. The average Tc value of BF of sprinters was significantly lower (19.5 ± 2.4 ms, $p<0.001$) than that of the same muscle in the sedentary group. There was a close relationship between the BF contraction times and the running speed on 20-m flying of the 15 young sprinters (Figure 2). The Pearson correlation coefficient was 0.72.

The test retest reliability showed that typical error of score difference (s=sdiff/√2, sdiff.) for MMG temporal parameters (Tc) was s=0.4 and the interclass correlation coefficient (ICC) was 0.99 (n=15). For 20m flying measurements (maximal speed test) the error/s was s=0.03 and ICC 0.92 (n=15).

Table 2: Contraction times (mean values ± SD) of the biceps femoris muscle in the group of young sprinters (n=15).

Subject	BF contraction time / ms
1	24.7
2	23.2
3	21.9
4	21.8
5	20.2
6	19.5
7	19.3
8	19.1
9	18.0
10	17.3
11	17.2
12	17.4
13	16.8
14	18.1
15	18.2
Average ± SD	19.5 ± 2.4

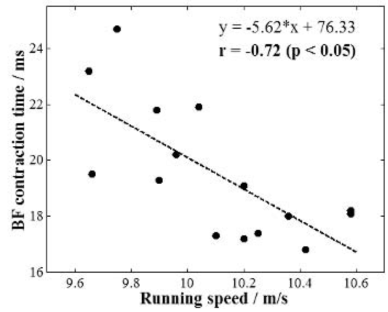


Figure 2: Correlation analysis between the maximum running speed (m.s-1) and contraction time (ms) of biceps femoris in the group of young sprinters.

IV. Discussion

In this study, using a non-invasive method of mechanomyography we hypothesize that biceps femoris muscle has a strong potential to transform in faster contracting muscle after long term sprint training. We presume that specific training process induced fiber type transitions from slow or type 1 fiber through intermediate type 2c fibers towards fast fiber types (2a and 2x) which resulted in shorter contraction time of BF. Though the fiber type transitions could not be directly evaluated with histochemical methods, as the muscle biopsies in sprinters were not possible due to ethical reasons, the results of the histochemical analysis of BF muscle, obtained by autopsy from presumably healthy untrained subjects, speak for a high adapting potential of this muscle. A relatively high proportion of intermediate type 2c fibers (5.7%), which co-expressed MyHC-1 and -2a, might represent a pool of fibers, capable of transformation either to slow type 1 or to fast type 2a according to the actual functional demands of the organism (11).

In BF muscle the percentage of type 2c fibers was higher than in any limb muscle of sedentary men, analyzed in our parallel study. A high percentage of type 2c fibers in BF muscle may also be considered as an indicator for the process of fiber type transitions in the muscle, as it was demonstrated that the percentage of hybrid fibers was drastically elevated in low-frequency-stimulated muscles. Similarly, an increase in the proportion of type 2c fibers was observed in elite runners after they changed from the endurance to sprint training (12). Therefore, according to the results of our study, a high proportion of type 2c fibers in BF muscle may also indicate that a limited transformation between type 1 and 2a fibers occurs under sedentary conditions and that 2c fiber constitute a transitional fiber type during the process of transformation.

Our results obtained by mechanomyography provided additional evidence that the BF muscle has a remarkable capacity to adapt to physical training. We demonstrated that the contraction time of BF was lower in the group of sprinters than in the sedentary group. Because of the strong correlation between the whole-muscle twitch contraction time and the percentage of slow twitch muscle fibers (4,5), we assume that fiber type transformation from slow to fast fiber types was achieved in the group of sprinters. Therefore, we can assume that the percentage of type 1 was lower in their BF muscle fibers than in the sedentary group. Furthermore, as fast fiber types predominate in sprinters (13), it can be hypothesized that the fiber type transformation under specific speed training proceeded from type 1 fibers through hybrid type 2c towards fast fiber types (2a and 2x) in the group of sprinters.

Our results also showed a strong correlation (r = 0.72) between the BF contraction time and the running speed on 20 m flying. On the contrary, in other measured muscles of sprinters, analyzed in our parallel study, no significant correlation between maximum speed and muscle contraction time was found, which implies that BF has an important functional role in sprinters. Our findings concerning the activity of BF are in agreement with the affirmation of

This page intentionally blank

reference (3) that the BF plays a primarily role in the propulsion phase of sprint running. On the other hand, reference (14) reported that the hamstrings muscles are not active during early swing, but in the late swing phase they decelerate hip flexion followed by controlling the knee extension. When the thigh begins its motion backward before the ground contact, the hamstrings act concentrically to extend the hip and flex the knee. The greatest and longer-lasting change in the muscle activity of the BF muscle with increasing speed may be responsible for further force production in the contact phase when extending the hip joint. The relative lengthening of the hamstrings activity during the stance phase with increasing running speed may emphasize the role of the hamstrings muscles to drive the body powerfully forward. In the same study (15) the finding that the push-off force in the horizontal direction greatly influences running speed gives further support for this interpretation. An increasing running speed requires an increased muscle activity of the leg extensors in the pre-activity and braking phases with longer-lasting increased activity of the hamstrings muscles. Nevertheless, we can conclude that the fastest sprinters had better adapted BF (shorter contraction time potentially as a consequence of higher percentage of fast twitch fibers) to develop maximum speed of running. But causality between shorter contraction time and higher adaptation ability of BF can not be related exclusively to the correlation. It seems that one of the necessary conditions for maximum speed development is the ability of biceps femoris to contract fast/have high percentage of fast twitch fibers; on the other hand in our study a similar relationship for other muscles was not established.

In conclusion, a high proportion of type 2c fibers in the group of sedentary men and lower contraction time of the muscle in sprinters in comparison with the sedentary group and high correlation between Tc and maximum speed (in-between sprinters) imply that BF muscle may have a high potential to tune its functional response with actual changes in physiological demands.

REFERENCES

[1] G.Goldspink, A. Scutt, P.T. Loughna, D.J. Wells, T. Jaenicke, and G.F. Gerlach, "Gene expression in skeletal muscle in response to stretch and force generation", *Am. J. Physiol.*vol. 262, pp. 356-63. 1992.

[2] D. Pette, and R. Staron, "Mammalian skeletal muscle fiber type transition", *Int Rev Cyt* vol. 170, pp. 143-223. 1997.

[3] N. Palastanga, D. Field, and R. Soames, *Anatomy and human movement.* In: Butterworth-Heinemann Ltd (eds) Oxford. 1989, pp. 331-335.

[4] R. Dahmane, Valencič V., Knez N. , and Eržen I., "Evaluation of the ability to make non-invasive estimation of muscle contractile properties on the basis of the muscle belly response" *Med. Biol. Eng. Com.* Vol**.** 38, pp. 51-56. 2001.

[5] R. Dahmane, Djordjevič S., Šimunič B., and Valenčič V., "Spatial Fiber Type Distribution in Normal Human Muscle. Histochemical and Mechanomyographical Evaluation" *J. Biomechanics* vol. 38/12, pp. 2451-2459. 2005.

[6] H Burger, Valenčič V., Kogovšek N., and Marinček Č., " Properties of musculus gluteus maximus in above knee amputees", *Clinical Biomechanics* vol.11(1), pp. 35-38. 1996.

[7] L. Guth, and Samaha F.J.. "Procedure for the histochemical demonstration of actomyosin MATPase (Research note)", *Exp. Neurol*. Vol. 28, pp. 365-67. 1970.

[8] F. Pernuš, Eržen I. , and Bjelogrlič Z., " A computer-aided method for muscle fiber type quantification", *Acta Stereol*. vol.5(1), pp. 49-54. 1986.

[9] E.F.Delagi, Perotto A., Iazzetti J, and Morrisson D., *The limbs. In: Anatomic guide for the electromyographer*. Charles C. Thomas, eds. Springfield Illinois USA, 1975. pp. 61-69.

[10] W.G. Hopkins, " Measures of reliability in sports medicine and science". *Sports Med*. vol. 30(1),pp 1-15. 2000.

[11] R. Billeter, Mac Dougall J.D., Bell R.D., and Howald H.," Myosin types in human skeletal muscle fibers", *Histochemistry* vol. 65, pp. 249-259. 1980.

[12] E. Jansson, Sjodin B., and Tesch P., " Changes in muscle fiber type distribution in man after physical training. A sign of fiber type transformation?" Acta Physiol. Scand. vol. 104, pp. 235-237. 1978.

[13] J.L Andersen, Schjerling P., and Saltin B.," Muscle, genes and athletic performance", *Sci. Am*. vol. 283, pp. 48-55. 2000.

[14] P.R Cavanagh,. *Biomechanics of Distance Running*. Champaign. In: Human Kinetics Books, 1990. pp. 53-56.

[15] H.P. Kyrolainen, et al.,"Changes in muscle activity patterns and kinetics with increasing running speed" *J. Strength Cond. Res.*vol.13 (4), pp. 400-406. 1999.

A Hybrid Data Transformation Approach for Privacy Preserving Clustering of Categorical Data

Dr.A.M.Natarajan	R.R.Rajalaxmi	N.Uma	G.Kirubhakar
Principal	Assistant Professor / CSE-PG	Final ME	Final ME
Kongu Engineering College	Kongu Engineering College	Kongu Engineering College	Kongu Engineering College
Perundurai	Perundurai	Perundurai	Perundurai
Erode, TamilNadu	Erode, TamilNadu	Erode, TamilNadu	Erode, TamilNadu
India	India	India	India
principal@kongu.ac.in	rrr_kec@yahoo.co.in	umamanivannan@yahoo.com	kirubhakar@rediffmail.com

Abstract - In today's information age there is a large availability of repositories storing various types of information about individuals. Data mining technology has emerged as a means of identifying patterns and trends from large quantities of data. The application of data mining technology to identify interesting patterns from these repositories leads to serious privacy concerns. Despite its benefit in a wide range of applications, data mining techniques also have raised a number of ethical issues. Some such issues are privacy, data security, intellectual property rights and many others. In this paper, we address the privacy problem against unauthorized secondary use of information. We focus primarily on privacy preserving data clustering on categorical data. In the proposed method, the categorical data is converted into binary data and it is transformed using geometric data transformation method. Then, clustering using conventional clustering algorithm is done on the transformed data to ensure privacy.

Index Terms: Clustering, Categorical data, Data Transformation, Binary data, Translation, Rotation, Scaling.

1. INTRODUCTION

The organizations collect data, often concerning individuals, and use them for various purposes, ranging from scientific research, as in the case of medical data, to demographic trend analysis and marketing purposes [1]. Organizations may also give access to the data they own or even release such data to third parties. The number of increased data sets that are thus available poses serious threats against the privacy of individuals and organizations.

Privacy preserving data mining (PPDM) is a novel research direction in data mining, where data mining algorithms are analyzed for side effects they incur in data privacy. The main consideration in privacy preserving data mining is two fold[1]. First, sensitive raw data like identifiers, names, addresses and the like should be modified from the original data base, in order for the recipient of the data not be able to compromise another persons privacy. Second, sensitive knowledge which can be mined from a database by using data mining algorithm should also be excluded, because such knowledge can equally well compromise data privacy. So, the main objective in PPDM is to develop algorithms for modifying the original data in some way, so that the private data and knowledge remain private even after the mining process [6].

In this paper, we focus on privacy preserving data clustering, notably when personal data are shared before clustering analysis. The idea is to partition a dataset into new clusters of similar objects. The goal is to group objects to achieve high similarity between objects within individual clusters and low similarity between objects that belong to different clusters.

To address privacy concerns in clustering analysis, we need to design specific data transformation methods that enforce privacy without loosing the benefit of mining. The key idea is to convert the categorical attribute value into binary value and it is transformed using geometric data transformation method. Then, clustering using conventional clustering algorithm is done on the transformed data to ensure privacy.

2. PRIVACY PRESERVING TECHNIQUES

There are many approaches adopted for privacy preserving data mining [4]. It can be classified based on the following dimensions:

- Data distribution
- Data modification
- Data mining algorithm
- Data or rule hiding
- Privacy preservation

The following sections discuss the approaches involved in each dimension.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 403–408.

2.1 Data distribution

Some of the approaches have been developed for centralized data, while others refer to a distributed data scenario. Distributed data scenarios can also be classified as horizontal data distribution and vertical data distribution. Horizontal distribution refers to these cases where different database records reside in different places, while vertical data distribution, refers to the cases where all the values for different attributes reside in different places.

2.2 Data modification

In general, data modification is used in order to modify the original values of a database that needs to be released to the public and in this way to ensure high privacy protection [4]. It is important that a data modification technique should be in concert with the privacy policy adopted by an organization. Methods of modification include:

•*Perturbation*, which is done by the alteration of an attribute value by a new value (i.e., changing a 1-value to a 0-value, or adding noise)

•*Blocking*, which is the replacement of an existing attribute value with a "?"

• *Aggregation* or merging which is the combination of several values into a coarser category

• *Swapping* that refers to interchanging values of individual records, and

• *Sampling* which refers to releasing data for only a sample of a population.

2.3 Data mining algorithm

This is actually something that is not known beforehand, but it facilitates the analysis and design of the data hiding algorithm. Various data mining algorithms have been considered in isolation of each other. Among them, the most important ideas have been developed for classification data mining algorithms, like decision tree inducers, association rule mining algorithms, clustering algorithms, rough sets and Bayesian networks.

2.4 Data or rule hiding

It refers to whether raw data or aggregated data should be hidden. The complexity for hiding aggregated data in the form of rules is of course higher, and for this reason, mostly heuristics have been developed. The lessening of the amount of public information causes the data miner to produce weaker inference rules that will not allow the inference of confidential values. This process is also known as "rule confusion".

2.5 Privacy preservation

This refers to the privacy preservation technique used for the selective modification of the data [4]. Selective modification is required in order to achieve higher utility for the modified data given that the privacy is not jeopardized. The techniques that have been applied for this reason are:

• Heuristic-based techniques like adaptive modification that modifies only selected values that minimize the utility loss rather than all available values

• Cryptography-based techniques like secure multiparty computation where a computation is secure if at the end of the computation, no party knows anything except its own input and the results, and

• Reconstruction-based techniques where the original distribution of the data is reconstructed from the randomized data. It is important to realize that data modification results in degradation of the database performance. In the proposed system, we use the data modification approach for privacy preservation.

3. Cluster Analysis

Clustering is an important data mining problem. The goal of clustering, in general, is to discover dense and sparse regions in a dataset. Most previous work in clustering focused on numerical data whose inherent geometric properties [2] can be exploited to naturally define distance functions between points. However, many datasets also consist of categorical attributes on which distance functions are not naturally defined. Recently, the problem of clustering categorical data started receiving interest.

3.1 Categorical Variables

Categorical variable (nominal variable) is a variable which can take more than two states and the domain of the categorical attribute is small [2]. For example, marital status is a categorical variable that may have, say three states: single, married, divorcee.

Let the number of states of the variable be M. The states can be denoted by letters or symbols. The dissimilarity between two objects i and j, defined by nominal variables can be computed using the simple matching approach:

$$d(i, j)=(p-m)/p$$

Where m is the number of matches (i.e., the number of variables for which I and j are in the same state), and p is the total number of variables.

3.2 Binary Variables

A binary variable has only two states: 0 or 1, where 0 means that the variable is absent, and 1 means that it is present. A binary variable is symmetric if both of its states are equally valuable and carry the same weight; that is there is no preference on which outcome should be coded as 0 or 1. One such example could be the attribute gender having the states male and female[2]. A binary variable is asymmetric if the outcomes of the states are not equally important, such as the positive and negative outcomes of a disease test. By convention, we shall code the most important outcome, which is usually the rarest one, by 1(e.g., HIV positive), and the other by 0(e.g., HIV negative).

4. The Proposed System

The proposed system consists of two steps. In the first step, the categorical attribute is converted into binary attribute. Then, in the second step the geometric data transformation technique is used to transform the binary data and any of the conventional clustering algorithms like k-means can be used for clustering. This data transformation technique ensures the privacy of the original data.

4.1 Converting categorical value into binary value

The Geometric data transformation methods can not be applied for the categorical value. So, it is to be converted into binary value. Categorical variable can be converted into asymmetric binary variable by creating a new binary variable for each of the M nominal states. For an object with a given state value, the binary variable representing that state is set to 1 while the remaining binary variable are set to 0.

PERSON	MARITAL STATUS
xxx	Single
yyy	Married
zzz	divorcee

⇩

PERSON	SINGLE	MARRIED	DIVORCEE
xxx	1	0	0
yyy	0	1	0
zzz	0	0	1

Figure. 1 Mapping Categorical to Binary value

For example, to encode the nominal variable marital status, a binary variable can be created for each of the three values listed in Fig. 1. For a person having the marital status ‘married’, the married variable is set to 1, while the remaining two variables are set to 0.

4.2 The Data Transformation Approach

4.2.1. The Basics of Data Perturbation

The methods based on the data perturbation approach fall into two main categories known as Probability-distribution category and Fixed-data perturbation category [3]. In the probability-distribution category, the security-control method replaces the original database by another sample from the same distribution or by the distribution itself. On the other hand, the fixed-data perturbation methods discussed in the literature has been developed exclusively for either numerical data or categorical data. These methods usually require that a dedicated transformed database is created for secondary use, and they have evolved from a simple method for a single attribute to multi-attribute methods. In all cases, such methods involve the addition of noise term with the mean 0, and hence result in no bias in estimating the mean. In this paper, we focus on fixed-data perturbation methods.

In its simplest form, fixed-data perturbation methods involve perturbing a confidential attribute X by adding some noise term e to result in the perturbed attribute Y. When this method is used for multi attribute databases, each attribute in the database is perturbed independently of the others. In general, this method is described as $Y = X + e$, where e is drawn from some probability distribution (e.g. Uniform, Normal) with mean 0 and a known variance to the data . These methods are referred to as Additive Data Perturbation (ADP). Apart from ADP methods, Multiplicative Data Perturbation (MDP) can also be used to provide aggregate statistics, while protecting the privacy of individuals represented in a database. In such a method, for a single confidential attribute X, the perturbed attribute Y is described as $Y = Xe$, where e has a mean of 1.0 and a specified variance. Since the mean of e = 1.0, there is no bias in estimating the mean. When the MDP method is used to distort multiple confidential attributes, each attribute must be perturbed independently of other attributes.

4.2.2. The Basics of Imaging Geometry

For the sake of simplicity, let us review the basics of imaging geometry in a 2D discrete space. However, the foundations are scalable to other dimensions [3]. In general a digital image a [m,n] described in 2D discrete space is derived from an analog image $a(x, y)$ in a 2D continuous space through a sampling process that is frequently referred to as digitization. The 2D continuous image $a(x, y)$ is divided into N rows and M columns. The intersection of a row and a column is termed a pixel. The value assigned to the integer coordinates [m,n] with $m = 0,1,2, ..M - 1$ and $n = 0,1, 2,…,N - 1$ is $a[m, n]$.

There are some transformations that can be applied to digital images to transform an input image a[m, n] into an output image $b[m,n]$. In this work, we consider the transformations translation, scaling, and rotation. We are expressing such transformations in a two-dimensional Cartesian coordinate system, in which a point has coordinates denoted (X,Y). The same transformations can be extrapolated to high dimensional data spaces.

Translation is the task to move a point within the coordinates (X,Y) to a new location by using displacements $(X0,Y0)$. The translation is easily accomplished by using a matrix representation $v' = Tv$, where T is a 2 X 3 transformation matrix depicted in Fig. 2a, v is the vector column containing the original coordinates, and v' is a column vector whose coordinates are the transformed coordinates. This matrix form is also applied to Scaling and Rotation.

Scaling by factors Sx and Sy along the X and Y axes is given by the transformation matrix seen in Fig. 2b. Rotation is a more challenging transformation. In its simplest form, this transformation is for the rotation of a point about the coordinate axes. Rotation of a point in a 2D discrete space by an angle θ is achieved by using the transformation matrix depicted in Fig. 2c. The rotation angle θ is measured clockwise and this transformation affects the values of X and Y coordinates.

$$\begin{bmatrix} 1 & 0 & X_0 \\ 0 & 1 & Y_0 \end{bmatrix} \quad \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \quad \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

(a) (b) (c)

Figure 2: Transformation matrices for (a) Translation (b) Scaling and (c) Rotation.

4.2.3. The Family of Geometric Data Transformation Methods

In this section, we introduce the family of geometric data transformation methods (GDTM) to meet privacy preservation in clustering analysis.

Basic Definitions

Let us assume that the data is represented as a matrix D_{mn}, where each of the m rows is an observation, O_i, and each observation contains values for each of the n attributes, A_i. The matrix D_{mn} may contain categorical and numerical attributes. However, our GDTMs rely on d numerical attributes, such that $d \leq n$. Thus, the m X d matrix, which is subject to transformation, can be thought of as a vector subspace V in the Euclidean space such that each vector $vi \in V$ is the form $vi = (a_1,\ldots, a_d)$, $1 \leq i \leq d$, where for all i ,a_i is one instance of Ai, $a_i \in R$ and R is the set of real numbers.

The vector subspace V must be transformed before releasing the data for clustering analysis in order to preserve privacy of individual data records. To transform V into a distorted vector subspace V', we need to add or even multiply a constant noise term e to each element vi of V. To do so, we define a *uniform noise vector* as follows:

Definition 1 (Uniform Noise Vector) Let N be a uniform noise vector, and for $1\leq i \leq d$, let $D_i(OP)$ be the set of operations associated with the domain of OP_i, and let $D_i(E)$ be the set of noisy term associated with the domain of NT_i.

The set of operations Di(OP) takes the values {Mult, Add, Rotate}, where Mult and Add correspond to a multiplicative and additive noise applied to one confidential attribute respectively. Rotate, denoted by $A_i \circlearrowright A_j$, implies that all instances of the attributes Ai and A_j are rotated by a common angle. In the next sections, we explain the use of the uniform noise vector N. Given the uniform noise vector N, we can transform the vector subspace V into the vector subspace V' by using a geometric transformation function.

Definition 2 (Geometric Transformation Function) Let V be a d-dimensional vector subspace, where each element v_i, $1\leq$ I $\leq$ d, is the form $v_i = (a_1,\ldots, a_d)$, and each a_i in v_i is one observation of a confidential numerical attribute, and let N be a *uniform noise vector. Geometric transformation function f can be defined as a bijection of d-dimensional space into itself which transforms V into V' by distorting all attributes of v_i in V according to its corresponding i-th element in N.*

In this paper, we consider the following geometric transformation functions: *Translation*, *Scaling*, and *Rotation* whose corresponding operations are *Add*, *Mult*, and *Rotate*. Based on the previous definitions, we can define a geometric transformation method (GDTM) as follows:

Definition 3(Geometric Data Transformation Method)

A geometric data transformation method of dimension d is a ordered pair, defined as GDTM = (V, f) where:

- $V \subset R^d$ is a vector subspace of data points to be transformed;
- f is a geometric transformation function, $f : R^d \rightarrow R^d$

The inputs for the GDTMs are the vectors of V, composed of confidential numerical attributes only, and the uniform noise vector N, while the output is the transformed vector subspace V'. All transformation data algorithms have essentially two major steps:

(1) Identify the noise term and the operation that must be applied to each confidential attribute. This step refers to the instantiation of the uniform noise vector N;

(2) Based on the uniform noise vector N, defined in the previous step, transform V into V' using a geometric transformation function.

4.2.4. The Translation Data Perturbation Method

In the Translation Data Perturbation Method, denoted by TDP, the observations of confidential attributes in each $vi \in V$ are perturbed using an additive noise perturbation [3]. The noise term applied to each confidential attribute is constant and can be either positive or negative. The set of operations $Di(OP)$ takes only the value {Add} corresponding to an additive noise applied to each confidential attribute. The sketch of the algorithm is given as follows:

Algorithm:

Input: V, N
Output: V'
Step 1. For each confidential attribute Aj in V, where $1 \le j \le d$ do
1. Select the noise term e_j in N for the confidential attribute Aj
2. The j-th operation $op_j \leftarrow$ {Add}
Step 2. For each $v_i \in V$ do
For each aj in $v_i = (a1,\ldots,ad)$, where a_j is the observation of the j-th attribute do
1. $a_{j'} \leftarrow$ Transform(a_j, op_j, e_j)
End

4.2.5. The Scaling Data Perturbation Method

In the Scaling Data Perturbation Method, denoted by SDP, the observations of confidential attributes in each $vi \in V$ are perturbed using a multiplicative noise perturbation. The noise term applied to each confidential attribute is constant and can be either positive or negative [3]. The set of operations $Di(OP)$ takes only the value {Mult} corresponding to a multiplicative noise applied to each confidential attribute. The sketch of the algorithm is given as follows:

Algorithm:
Input: V, N
Output: V'
Step 1. For each confidential attribute Aj in V, where $1 \le j \le d$ do
1. Select the noise term e_j in N for the confidential attribute Aj
2. The j-th operation $op_j \leftarrow$ *{Mult}*
Step 2. For each $v_i \in V$ do
For each a_j in $v_i = (a_1, \ldots, a_d)$, where a_j is the observation of the j-th attribute do
1. $a_{j'} \leftarrow$ Transform(aj, opj, ej)
End

4.2.6. The Rotation Data Perturbation Method

The Rotation Data Perturbation Method [5], denoted by RDP, works differently from the previous methods. In this case, the noise term is an angle θ. The rotation angle θ, measured clockwise, is the transformation applied to the observations of the confidential attributes. The set of operations $Di(OP)$ takes only the value {Rotate} that identifies a common rotation angle between the attributes Ai and Aj.

Unlike the previous methods, RDP may be applied more than once to some confidential attributes. For instance, when a rotation transformation is applied this affects the values of two coordinates. In a 2Ddiscrete space, the X and Y coordinates are affected. In a 3D discrete space or higher, two variables are affected and the others remain without any alteration. This requires that one or more rotation transformations are applied to guarantee that all the confidential attributes are distorted in order to preserve privacy. The sketch of the algorithm is given as follows:

Algorithm:

Input: V, N
Output: V'

Step 1. For every two attributes A_j, A_k in V, where $1 \le j \le d$ and $1 \le k \le d$ do
1. Select an angle θ for the confidential attributes Aj, Ak
2. The j-th operation $op_j \leftarrow$ {Rotate}
3. The k-th operation $op_k \leftarrow$ {Rotate}
Step 2. For each $vi \in V$ do
For each al in $vi = (a1,\ldots,ad)$, where al is the observation of the l-th attribute do
1. $aj' \leftarrow$ Transform(a_l, op_l, e_l)
End

4.2.7. The Hybrid Data Perturbation Method

The Hybrid Data Perturbation Method, denoted by HDP, combines the strength of the previous methods: TDP, SDP and RDP [3]. In this scheme, we select randomly one operation for each confidential attribute that can take the values {Add, Mult, Rotate} in the set of operations $Di(OP)$. Thus, each confidential attribute is perturbed using an additive, a multiplicative noise term, or a rotation.

The sketch of the algorithm is given as follows:

Algorithm:

Input: V, N
Output: V'
Step 1. For each confidential attribute A_j in V, where $1 \le j \le d$ do
1. Select the noise term e_j in N for the confidential attribute Aj
2. The j-th operation $opj \leftarrow$ {Add,Mult,Rotation}
Step 2. For each $v_i \in V$ do
For each aj in $v_i = (a_1,\ldots,a_d)$, where aj is the observation of the j-th attribute do
1. $aj' \leftarrow$ Transform. (a_j, op_j, e_j)
End

This hybrid data transformation approach can be used to transform the binary data to preserve the privacy of the original data. The conventional clustering algorithms like k-means can then be applied on the transformed data and we ensure that the clustering results will be the same.

5.CONCLUSION

The privacy preserving data mining is used to preserve data and knowledge even after mining. Privacy preserving clustering on categorical data is a difficult task. In the proposed system, the categorical data is converted into binary data in order to apply the geometric data transformation. Then , it is transformed to some other format using GDT before data mining. Any of the conventional clustering algorithms for numeric data can then be used for clustering the transformed data. We ensure that by using this method we can preserve the original data and also the clustering accuracy.

6. REFERENCES

[1] Elisa Bertino, Ravi Sandhu," Database Security - Concepts, Approaches, and Challenges", IEEE transactions on dependable and secure computing, vol. 2, no. 1, January-March 2005

[2] J.Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, CA,2001.

[3] Stanley R. M. Oliveira, osmar R. Zaiyane ,"Privacy Preserving Clustering by Data Transformation". In proceedings of 18th Brazilian Conference on Databases,2003

[4] V.S. Verykios, E. Bertino, I.N. Fovino, L.P. Provenza, Y. Saygin, and Y. Theodoridis, "State-of-the-Art in Privacy Preserving Data Mining," ACM SIGMOD Record, vol. 3, no. 1, Mar. 2004.

[5] K. Chen and L. Liu. Privacy preserving data classification with rotation perturbation. In Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM'05), Houston, TX, November 2005

[6] R. Agrawal and R. Srikant, "Privacy Preserving Data Mining," Proc. ACM SIGMOD Conf. Management of Data, May 2000

ANUPAM – Ameya: A Teraflop Class Supercomputer

Rajesh Kalmady, Vaibhav Kumar, Digamber Sonvane, Kislay Bhatt, B.S.Jagadeesh, R.S.Mundada, A.G.Apte, P.S.Dhekne

Computer Division, Bhabha Atomic Research Centre,
Trombay, Mumbai, India

Abstract- **High Performance Computers are essential tools of trade in modern day science. The current research problems scientists and engineers are attempting to solve are becoming increasingly complex and are beyond the reach of traditional laboratory style experimental methodologies. Access to supercomputers makes a great deal of difference as they help in dealing with very high resolution models which result in accurate simulations of the phenomena that one is trying to investigate, in reasonable timeframes. This, in turn helps in cutting down the number of experiments that one has to conduct and also in arriving at accurate inferences at a faster rate. The Computer Division of the Bhabha Atomic Research Centre (BARC) has been developing high performance parallel computers for its scientists and engineers for over a decade. The latest in the ANUPAM series of parallel supercomputers is called 'Anupam-Ameya', a 512-processor cluster running Linux. This paper describes the design goals, architecture, components and benchmarking figures of the Anupam-Ameya cluster.**

I. Introduction

"Out-compute to Out-compete" is the dictum of modern science. In order to carry out frontline research and development and remain internationally competitive, it is extremely important that scientists have access to high performance computers. BARC is a premier research organization working on the development of technologies related to atomic energy and its applications in a wide range of areas. BARC scientists are engaged in research in various fields of science such as physical sciences, chemical sciences, nuclear and atomic sciences, biology and engineering, using computers extensively to meet their requirements of high performance computing, scientific computing, visualization, information processing and information exchange. It is extremely important that the computing and communication facilities available at BARC are not only enough to meet the growing requirements but also comparable with the best.

We, at Computer Division, BARC have been striving towards meeting these requirements of users by providing advanced computing facilities on par with the rest of the world. As part of this programme, the ANUPAM series of parallel computers developed by BARC have been the computational workhorses for BARC users for over a decade. Our major efforts have essentially been in developing systems, software and applications based on open source platforms and are based on off-the-shelf commodity components.

So far, BARC has developed sixteen different models of the ANUPAM series using a variety of processors as compute nodes and various interconnection technologies. We started initially with parallel processing systems based on Intel i860 processors and Multibus-II interconnect, progressing finally to Linux clusters using Intel x86 processors and Gigabit Ethernet interconnect. The performance figures of these machines have gone up from a modest 30 MFLOPS in 1991 to 365 GFLOPS in 2003 and finally to 1.73 TFLOPS in 2005. The number of processors has also increased from 4 to 512 over the years.

The latest in the ANUPAM series and the first teraflop class machine is called Ameya. This is a Linux based cluster consisting of 512 processors, connected over a Gigabit Ethernet network.

II. Design Goals

Apart from providing raw computing power, the design goals of the ANUPAM series of parallel computers have also been

- To maximize the use of open standards and technologies
- To harness commodity components
- To establish that a general purpose architecture is able to cater to a wide variety of problems

These goals ensure that the time required to build a new machine is as short as possible and also keep the cost down. BARC being a multi-disciplinary research centre, it would not be feasible to cater to a specific subset of problems and build a system optimized for that class. Hence, the machine should be able to support all kinds of compute models and messaging patterns equally well.

This was the philosophy on which all ANUPAM systems were based. Starting with a bus based architecture with MultiBus-II and Intel i860 processors, we later on adopted the cluster of workstations architecture, which rose in popularity during the mid-nineties with the advent of high speed networking technologies. The rise in performance of Intel processors and emergence of Linux as a robust and stable

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 409–414.

operating system made this architecture the platform of choice for high performance computers.

Building Linux clusters for HPC applications is now a well-understood concept and plenty of expertise and literature is available for doing this[1]. However, building a large cluster consisting of hundreds of nodes is a different ballgame altogether. Along with issues of performance and scalability, there are issues like stability, availability, cluster monitoring and management to worry about. As a large cluster, the system should be able to handle a large number of jobs and users and thus should be designed to be efficiently managed. Large clusters also have issues such as optimization of space, layout, power and cooling, each of which has to be addressed properly.

III. Architecture And Design Rationale

The ANUPAM Ameya system is based on the concept of Cluster of Workstations. It is a compact, centralized, homogenous Linux cluster of 256 dual processor nodes interconnected by Gigabit Ethernet network. A logical view of ANUPAM Ameya architecture is described in figure 1 below.

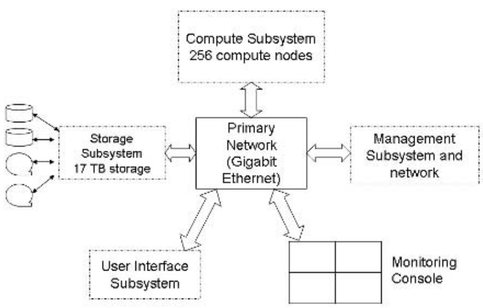


Fig. 1: Block Diagram of ANUPAM-Ameya Architecture

The major subsystems, key components and their design rationale are explained below:

A. *Compute Subsystem*

The Compute Subsystem consists of compute nodes and the software providing the parallel programming environment.

Compute Nodes**:** There are 256 compute nodes, which are rack mount servers of 1U size. Each of the compute nodes consists of two Intel Xeon EM64T processors @ 3.6GHz with 2MB cache each, shared 4GB memory and a SATA disk of 80GB capacity. In addition to the above-mentioned compute nodes, there are nine spare nodes in the system in order to increase the availability.

The choice of compute node was made after extensive evaluation of various models. The selection criteria for compute node models were performance, stability, acoustic noise, cooling and management features. The performance of various models was evaluated using a set of benchmark programs that evaluated the machine in different areas (such as network, memory, CPU and so on). Stability of the machine was tested by keeping the machine under the test-procedure for 3 days in continuous operation and checking the performance. The machine was deemed to have failed if the performance degraded with time or if the test failed. Some of the features like BIOS support for remote installation and console redirection to the serial port required for cluster installation and management were also tested.

Software**:** The Operating System on all the nodes is Scientific Linux 4.1. Parallel programming environment is provided by MPICH, LAM MPI, PVM and ANULIB libraries. Compilers for FORTRAN, C, C++ and various scientific libraries such as BLAS, LAPACK, SCALAPACK, etc. are also available.

B. *Storage Subsystem*

The storage subsystem consists of file servers, backup servers and tape libraries.

File Servers**:** There are 12 file servers, which constitute 17 terabytes of storage space. Each server is a 2U rack-mount server, equipped with dual Xeon processors.

Backup Servers**:** There are two backup servers for taking backup of users' data. Each server is a 5U rack-mount server equipped with dual Xeon processors and backup devices such as DVD writer, DAT and DLT drives. In addition, each server is connected to a tape library. The backup servers offload the backup load from the main file servers.

Tape Libraries**:** There are two tape libraries with a storage capacity of 3.2 terabytes each. The backup servers are programmed to take periodic backup of file servers to their local disks and then copy it onto the tape libraries.

Design challenges for the storage subsystem were performance, reliability and availability. Necessary redundancy is provided to reduce failures and downtime. Moreover, the system design ensures minimal effect of a server failure. RAID is configured on each server to overcome single disk failures. Each server has three Gigabit Ethernet network ports, which are link-aggregated to increase the availability and throughput by three fold. The users are distributed across the 12 file servers so that a single server failure affects only a fraction of users while others can still use the system.

C. *Primary Network*

The primary network in our cluster is Gigabit Ethernet The same network is used for inter-processor communication as well as NFS file server traffic. To decide upon the optimal values for network parameters such as TCP window size and frame size, different combinations of these parameters were tried out and the bandwidth was measured with NetPIPE [2]. Based on the results of this experiment, the frame size was set to Jumbo Frames of 3000 Bytes and TCP window size was set to 5 MB.

To achieve high performance it is essential to choose an interconnect network with high bandwidth and low latency. There was a choice of two interconnects viz., Gigabit Ethernet and SCI (Scalable Coherent Interface) networks. The performance of these two networks were evaluated by extensive benchmarking and it was found that though SCI fared better than Gigabit Ethernet in all the tests, it made a significant difference only in communication intensive applications [3]. Thus, Gigabit connectivity was preferred as our applications were disparate in nature, having a mix of compute and communication intensive portions. Moreover, with Gigabit Ethernet, it was much easier to manage the cabling complexity and it was also observed that the system stability was better with Gigabit Ethernet.

D. Management Subsystem and Network

Management subsystem needs to be independent of storage and compute subsystems. It needs to be robust and always available. Other major challenge is to make the system manageable remotely. Our Management Subsystem consists of dedicated Service Nodes, Terminal Servers, PDUs, and a separate Management Network.

Service Nodes:

There are service nodes configured to run various service functions in the system. The functions of various service nodes are as given below:

- Head node: This is the gateway for outside world to access the cluster. Users log into this node and compile or submit their jobs.
- Cluster Monitoring and Management using ANUNETRA Monitoring System
- Accounting System and Database
- Torque Queue Manager
- Installation and Configuration Manager
- Mail gateway

In small cluster these service functions may well reside on a single machine. In order to distribute the load and minimize the effect of failing service nodes, it was decided to have dedicated nodes for various services. Moreover, there are standby nodes configured for critical services to enhance the availability of the system further.

Terminal Servers and PDUs**:** Terminal Servers and network controlled Power Distribution Units (PDUs) are used for remote management of the system. Each node’s console is redirected to its serial port for remote access. These serial consoles are accessed using terminal servers by reverse telnet. PDUs are used to control the power to the nodes through software, i.e. for starting in a sequence, for selectively powering ON/OFF and for rebooting the nodes.

Management Network: A dedicated management network of Fast Ethernet Switches connects the Terminal Servers and PDUs. This network is connected to the primary network using a Layer 3 switch and VLAN technology.

E. User Interface Subsystem

The system is connected to the intranet and users can log into the head node to compile or submit their jobs. There are also dedicated user terminals connected to the system.

F. Monitoring Console

There is a separate console machine with quad display card connected to 2x2 tiled LCD screens. This runs the ANUNETRA monitoring application, which displays multiple system parameters on multiple screens and thus acts as a monitoring console for operators.

IV. LAYOUT

The layout of ANUPAM-Ameya system is designed for optimal space utilization with adequate space allocated for proper air circulation and cabling. The cables are laid such that future repairs and maintenance can be carried out easily.

The entire system consisting of 275 nos. of 1U servers, 12 nos. of 2U file servers, switches and other paraphernalia is housed in 14 nos. of 42U racks. Ten ‘compute’ racks house the 256 compute nodes, two racks house the storage subsystem and the rest two house the service nodes, spare nodes and the network switches. Even though the compute racks can hold 42 1U servers, only 25 or 26 of them are mounted in each rack in order to alleviate problems of power supply, cabling and cooling. Each compute rack also houses two 16 port terminal servers connected by serial cables to the serial consoles of each compute node. There are also four network controlled 8-outlet power distribution units in each rack which power all the servers contained therein. These PDUs are programmable and they have been configured to power the outlets in sequence to avoid tripping of the mains due to heavy inrush current during the power-on process.

The Gigabit Ethernet switches comprising the compute network are housed in the central racks for the sake of symmetry and to avoid lengthy cables. Different colored cables are used for the Gigabit Ethernet, Fast Ethernet and serial cables for ease of identification and maintenance. Spare Gigabit Ethernet and serial cables are run into each rack for the sake of redundancy.

Two racks house the storage subsystem, which comprises of 12 nos. of 2 U file servers, 2 backup servers and tape libraries for backup. Since each of these components has multiple redundant power supplies, care is taken to see to it that this feature is fully made use of. Each redundant power supply of each server is powered from a different PDU taking power from different UPS. This is to ensure availability of the server even on failure of a power supply unit, PDU, UPS or the mains. All critical components of the system such as the interconnection network, storage subsystem, management nodes and monitoring consoles are powered through UPS.

Clusters built out of 1U servers generally end up with a large number of servers mounted in a single rack. Such high density clusters need to be cooled properly so that adequate cold air reaches inside the narrow chassis of the 1U servers. For Ameya, cold air is provided in the front of the rack both from the ceiling ducts and through the false flooring. This creates a wall of air in front of the racks and this air is sucked into the servers by the CPU fans. Hot air is blown to the rear of the racks where heavy-duty fans pull it out and direct it to the vents in the ceiling. This kind of directed airflow ensures that the processor temperature is kept within acceptable limits.

V. Cluster Management

A large cluster such as Ameya has to be efficiently managed in order to function as a centralized computational resource for a large population of users. ANUPAM's cluster management suite consists of a set of both in-house developed tools and open-source management software. The management functions start with the installation and configuration of the system and include run-time monitoring and management, health checking, troubleshooting, job management and accounting.

A. *AnuInstall:*

Anuinstall is the installation tool for the Ameya cluster. Anuinstall coupled with the configuration manager automates installation and post-installation tasks.

Ameya has different nodes configured to run different set of services, which makes their installation and configuration difficult to manage. Anuinstall has an upper hand over other installation tools such as OSCAR and Rocks in two ways [4] [5][6].

Firstly, other installation tools consider just two classes of nodes in a cluster viz., server and compute node. They install a single server running all the services and identical compute nodes having same configuration. Anuinstall does not have any server or node class. Instead, it uses the configuration manager to find the set of software packages to be installed on a particular node.

Secondly, Anuinstall, like all other cluster installation tools uses network installation feature of Linux using PXELINUX [7]. However, it does not keep a separate copy of the kickstart file for each node; instead it generates the kickstart file of the node by a CGI script based on the node's IP address and Hostname. The kickstart file is requested using HTTP and the web server runs CGI scripts to generate the kickstart file based on the requesting IP and configuration stored in database. Based on the IP address, hostname is resolved and on basis of the hostname, the closest matching template of options is selected. This template contains the details of the operating system version, distribution, list of software packages to be installed and services to be run on the node. At the end of the installation, a log file of installation is uploaded on the central server for checking successful installation. An email is also sent to the administrator when installation is completed.

Installation of new nodes is little bit tricky because the mapping of the node's MAC address to its IP address is not known. For this, installation is done in two steps; in the first step a minimal OS installation is done and hostname to MAC address mapping is registered and in the second step complete OS with full configuration is installed.

B. *Configuration Manager*

The Configuration Manager is tightly coupled with the installation system; it uses the same database to maintain consistency of configuration data. A configuration-free client is installed on each node, which searches for the configuration server using broadcast. The client downloads a node-specific list of packages and files from the configuration server. It then checks the installed packages and files with the required ones and installs missing components, if any. A report of mismatch, in the required and existing configuration, is sent back to the server. The server stores the report in the database and sends a mail to the Administrator.

For scalability and robustness of the system, the client communicates with the server pseudo randomly. It waits for a random amount of time that is approximately equal to the requested interval to contact the server. This eliminates the possibility of the server being flooded with a large number of requests at a particular point of time and remaining idle for rest of the time. Moreover, to reduce broadcast traffic the client remembers the address of the last server and tries to contact it. If the server does not respond, the client sends out a broadcast to find out the address of the new server.

C. *Anunetra*

Anunetra is a cluster monitoring and management system. It is developed to increase the availability of the nodes, report errors, generate alerts, and to provide a centralized management. It provides centralized interface to perform health check activities in the cluster and some basic management tasks.

Anunetra continuously monitors Ameya system and reports errors in case of deviation from normal operating values. Metrics like cpu temperature, fan speed, disk usage, cpu usage, cpu count, cpu speed, swap used, memory free, bytes in/out, packets in/out etc. are under continuous monitoring. Its management interface, with authentication, allows the administrator to perform some common tasks (offline/online of nodes, network connectivity check, services check, available disk space etc.) on multiple nodes from a single interface (without logging into each node).

Anunetra uses publisher and transport mechanism of Ganglia and the other modules such as collection, archiving and presentation are designed as per our needs and constraints [8][9]. Other than common monitoring features, it incorporates facilities such as Reports, Analysis, Alert System, Auto Restart of failed nodes, Resource and Job Information, Login on Click, Management Interface and so on.

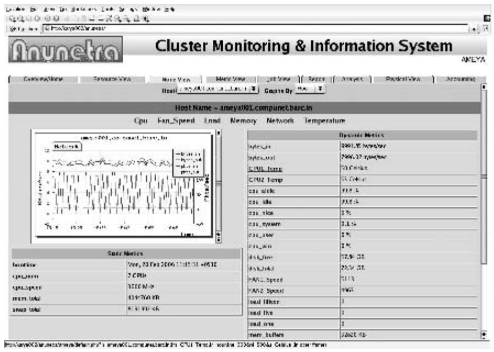


Fig. 2: Anunetra Cluster Monitoring System

D. Serial console logger

Serial Console Logger logs all the text written by a node on its console. This text is mainly status and warning/error messages written by the kernel and its modules. Each node's console is redirected to its serial port that is connected to a terminal server. This converts the console of a node into a network entity, which can be accessed remotely. These network entities are continuously polled to collect console messages, which are then written to the respective files.

Equipped with a configuration file for conditional logging, it extracts connection information (nodes to terminal server), makes connections to the ports and logs all the data written. To provide access to the console, it runs as a service and listens to TCP port 4000. It allows access to the console of any connected node using telnet client program.

E. Job Management

Job Management System on ANUPAM-Ameya, is based on TORQUE Resource Manager and MAUI Job Scheduler [10][11][12]. These tools are complemented with some in-house code developed as per our needs. User jobs have different types of resources requirements, e.g. some jobs need large number of processors whereas some saturate at small number of processors, some jobs run for one or two days whereas some go up to a week, some jobs have to run on some subset of nodes due to software licenses binding and so on. Considering all these requirements, three job queues are provided with different scheduling and execution policies for each. In addition, a test queue, with no wait time, is also provided where users can test their jobs for error-free execution before submitting them to one of the production queues.

Scheduling Policy for jobs in Ameya is optimized based on demand and available resources. It is designed to avoid starvation of large jobs and ensure better utilization of resources. Throttle limit for scheduling maximum jobs per user at a time is set and a fair-share policy provides a fair portion of the cluster's resources to each user under heavy load conditions. The scheduler schedules two jobs per user at a time, but during heavy load, it dynamically decides number of jobs to be run per user based on historical data of previously executed jobs.

Given a schedule with advance-reserved high-priority jobs and a list of low-priority jobs, the scheduling algorithm tries to fit the small jobs into scheduling gaps. This allocation does not alter the sequence of jobs previously scheduled, but improves system utilization by running low priority jobs between high priority jobs. This gives large jobs a guaranteed start time, while providing a quick turn around for small jobs.

F. Resource Reservation System

Provision of Resource Reservation is also available in Ameya system. There are some events for which advance reservation of resources need to be created. For example,

- A user may request for urgent job-execution
- A user from other organization may arrive for testing or demonstration
- Resources may be needed for maintenance purpose

Resource Reservation System (RRS) provides a web-based GUI to the administrator to create and release reservations. There is a provision to supply name of the user, number and type of nodes to be reserved, date and time when the reservation should start and the duration of the reservation. After submitting this information, the reservation is created for the user and details are shown in the form of a table.

G. Parallel Shell

Users are not allowed to login directly or to run remote shell on any of the compute nodes. However, sometimes it is necessary for the users to run some shell commands on the compute nodes. Therefore, a Parallel Shell was developed, through which users can execute some restricted shell commands, remotely, on the compute nodes. By default, a command runs on all the nodes, but, if the users want to run the commands on selected nodes, then they need to supply node-names along with the commands.

H. ANUPAM Accounting Package

Anupam Accounting Package (AAP) provides various details about Ameya System, e.g. system utilization, users' statistics, present status of the nodes and so on. It has a centralized database that records the information about users' jobs. The database is populated by data-entry programs, installed on all the nodes, which are invoked on submission, startup and termination of jobs. Job parameters such as username, number of nodes, node-names, job start and end times and so on. are fetched by the programs and entered into the database.

An interactive web interface is available, as a front end, for presentation of the collected data. Using the web-interface, one can extract and display the following information in tabular as well as graphical form.

- Daily/monthly report of utilization of nodes and system
- Daily/monthly report of users' statistics and their system utilization
- Job on a node at a particular time
- Average waiting time of jobs in the queue

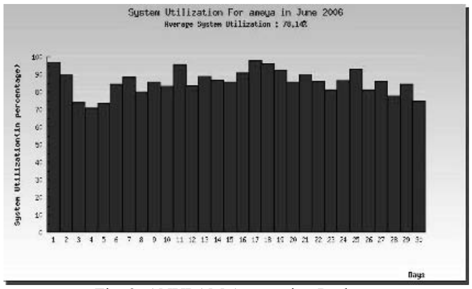


Fig. 3: ANUPAM Accounting Package

VI. BENCHMARKING

The ANUPAM Ameya system was benchmarked using the standard High Performance Linpack program used in the Top500 ratings. This program requires an MPI library and an optimized BLAS library. The BLAS library developed by Dr. Kazushige Goto, commonly known as the Goto BLAS was used in the benchmark runs [13]. Goto BLAS is generally used in most of the top500 rating runs because of its high degree of optimization, which helps to obtain best possible HPL ratings.

The best obtained HPL reading on the Ameya system was 1.73 TFLOPS out of a peak of 3.6 TFLOPS with a matrix size of 316000. This works out to be about 50% of the peak, which seems to be the best possible performance obtainable with Gigabit Ethernet interconnect and open source software components.

The following table shows the different HPL readings obtained with different numbers of nodes. It is known that the best HPL ratings are obtained when the matrix size is such that it occupies 80% of each node's memory. Therefore, the problem size is increased with increasing number of nodes, starting with 20000 for one node to 316000 for 256 nodes.

TABLE 1
HIGH PERFORMANCE LINPACK READINGS

No. of processors	Matrix Size	HPL Ratings (GFLOPS)
2	20000	12
4	28000	22
8	40000	42
16	56000	82
32	80000	155
64	112000	277
128	160000	526
256	224000	914
512	316000	1730

VII. CONCLUSIONS

The ANUPAM-Ameya parallel system is under operation for almost a year now. With about a 100 users, it has taken over the load of the other older and smaller clusters in BARC and has quickly become the main workhorse of computing for BARC's scientists. The utilization figure of Ameya over the past several months is around 75%.

Designing a large cluster is not simply a matter of scaling up the components used in building small clusters. Large clusters have their own specific problems that need to be attended to. The experience gained in Ameya's design and implementation has given us a better understanding of the issues involved and put us on the path towards developing bigger and faster clusters in the future.

ACKNOWLEDGMENT

It is our pleasure to acknowledge the encouragement and constant support received from Mr. G. Govindarajan, Former Director, E&I Group, BARC and Dr. S. Banerjee, Director, BARC. In addition, we gratefully acknowledge the untiring efforts of the technical staff of Computer Division who assisted us in building this cluster and making it operational in a span of 2 months.

REFERENCES

[1] Dr. Andreas Koch, "High Performance Computing Cluster", http://www.transtec-cluster.com/doit/loadva/software/w3/ INV_HOME_HTML/ whitepaper/sys-2003-3_hpc_e.pdf
[2] NetPIPE: A Network Protocol Independent Performance Evaluator http://www.scl.ameslab.gov/netpipe/
[3] Rajesh Kalmady, Digamber Sonvane, "Performance Evaluation of Gigabit Ethernet and SCI in a Linux Cluster," Proceedings of the First International Conference on Distributed Computing and Internet Technology, ICDCIT 2004, Springer Berlin, Volume 3347/2004, pp 50-59.
[4] Mattson T.G., "High Performance Computing at Intel: the OSCAR software solution stack to computing", First IEEE/ACM International Symposium on Cluster Computing and the Grid, 2001, pp22-25
[5] Rocks, http://www.rocksclusters.org/
[6] Emir Imamagić, Davor Mihajlović, "Automatic Installers Review", 6th Carnet Users Conference, Zagreb, Sep 2004
[7] PXELINUX - SYSLINUX for network boot http://syslinux.zytor.com/pxe.php
[8] Ganglia Monitoring System http://ganglia.sourceforge.net
[9] Sacerdoti, F.D. Katz, M.J. Massie, M.L. Culler, D.E. "Wide area cluster monitoring with Ganglia", Proceedings.of the IEEE International Conference on Cluster Computing, 2003
[10] Cluster Resources - TORQUE Resource Manager http://www.clusterresources.com/pages/products/torque-resource-manager.php
[11] Cluster Resources - MAUI Cluster Scheduler http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php
[12] David Jackson, Quinn Snell, Mark Clemenet, "Core Algorithms of the Maui Scheduler", Proceedings of the 7th International Workshop on Job Scheduling Policies for Parallel Processing, June 2001, Page 87.
[13] High Performance BLAS by Kazushige Goto, http://www.tacc.utexas.edu/~kgoto/

Dealing with Concurrent Regions during Scenario Generation from Activity Diagrams

Robert Chandler, Chiou Peng Lam, and Huaizhong Li
Edith Cowan University
Bradford Street, Mount Lawley,
Western Australia 6050
(robert.chandler; p.lam; @ecu.edu.au – huaizhongli@ieee.org)

Abstract- ***Scenarios are a popular focus for the acquisition and validation of system requirements and in the generation of system-based test-cases. However, generating scenarios manually is a tedious process, which may introduce errors or produce incomplete scenario sets. This paper discusses an approach to scenario capture, in support of requirements engineering that can then be used for test-scenario capture and test-case generation. The approach has been automated successfully, producing usage-scenarios from UML (Unified Modelling Language) Activity Diagrams without the need for manual intervention. The paper walks through the approach - with a specific focus on the processing of concurrent regions - and compares the results with other approaches applied to the same UML activity model.***

I. INTRODUCTION

Scenarios are a helpful means of identifying and communicating system requirements [5], for verifying the completeness and correctness of a functional model [11], [23] and for aiding in the generation of test-cases [1], [17], [25] and [26]. Unfortunately, it is an extremely labour intensive process to capture them manually [24] and according to a recent study [22], there is a distinct lack of both methods and tools to support the seamless progression from modelling tool to Usage Scenarios (US), without the need for manual intervention.

Existing methods tend to use some form of manual derivation [3], [21], which may introduce errors or produce incomplete scenario sets [10]. Still other methods employ graphical scenario representations, such as Petri-nets [12], [15] or Message Sequence Charts [2], [14], [16] that do not form any part of the alleged de-facto standard for software development; the UML [4], [6], [9], [20].

This paper proposes an approach that uses the information exported from a UML model, in XMI (eXtensible Mark-up Language (XML) Metadata Interchange) format, to produce USs from Activity Diagrams (AD) that the model contains. This approach processes complex sequences of AD artefacts, that are regularly found within concurrent regions, which are often neglected in other usage-scenario generation approaches.

Activities that have looping or skipping paths make scenario capture difficult; this difficulty is compounded when looping or skipping occurs within a concurrent region; and more-so when in nested concurrent regions. Some approaches have also viewed the inclusion of multiple final and/or initial nodes in an AD as a difficult factor in the capture of scenarios [25], [26]; whereas this approach does not. The UML 2.0 specification [20] for AD modelling provides for the inclusion of multiple initial and final nodes at the basic-level of AD modelling. The approach discussed in this paper has been applied to models at both the Complete-Structured and the Intermediate levels for software development and process modelling respectively.

The rest of this paper is organized as follows. ADs are defined more formally in section II. The comparative approaches are introduced in section III along with some background on this area of research. We introduce the approach and algorithm in section IV and a description of the sequences of artefacts found when dealing with concurrency in ADs in section V. Section VII introduces the case-study/UML activity model upon which the approaches are applied and sub-section VII.A details the criteria used for comparing the processing results; which are presented in sub-section VII.B. Finally, our conclusions are drawn in section VIII.

II. AD DEFINITION

Typically, the modelling of an activity will begin with an initial node and progress through a set of edges and nodes until the activity ends at one or more final or flow-final nodes. We offer the following definition to formalize the structure of an AD:

$AD = \{V, E\}$

Let AD be an Activity Diagram within the set of ADs in a UML Model; furthermore, let:

V be a set of Artefacts where:

$A = \{a_1, a_2, \ldots a_n\}$ is a finite set of action elements;
$B = \{b_1, b_2, \ldots b_p\}$ is a finite set of branches;
$F = \{f_1, f_2, \ldots f_q\}$ is a finite set of forks;
$I = \{i_1, i_2, \ldots i_r\}$ is a finite set of initial nodes;
$J = \{j_1, j_2, \ldots j_s\}$ is a finite set of joins;
$M = \{m_1, m_2, \ldots m_u\}$ is a finite set of merges;
$O = \{o_1, o_2, \ldots o_v\}$ is a finite set of objects;
$Z = \{z_1, z_2, \ldots z_w\}$ is a finite set of finals or flow-final nodes;
$E = \{e_1, e_2, \ldots e_x\}$ be a finite set of edges.

Based on these sets of artefacts, an AD consists of sequences of vertices and edges that are connected according to the semantic rules established in the UML specifications.

$AD = \{i_n, e_1, (a_1 \vee b_1 \vee f_1 \vee m_1 \vee o_1 \vee z_1), e_2, (a_2 \vee b_2 \vee f_2 \vee j_2 \vee m_2 \vee o_2 \vee z_2), \ldots e_n, (a_n \vee b_n \vee f_n \vee j_n \vee m_n \vee o_n), e_{n+1}, z_n)$

To direct the flow of control and data throughout an activity, decision edges may carry guard conditions to determine which outgoing edge will receive 'control' to continue the activity. We define guard conditions thus:

T. Sobh (ed.), *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, 415–420.

`G = {g1, g2, ...gx} is a finite set of guard conditions;` and

G_x is in the corresponding edge e_x such that:

for all G_x there exists e_x

III. RELATED WORKS

The US Department of Defence (DoD) prepared a guidebook [18] defining a set of procedures to establish a structured and disciplined approach to integration testing from end-to-end (E2E). The E2E guidebook describes usage scenarios as thin-threads; stating that a thin-thread is a complete scenario from the end-user's point of view. A thin-thread describes just one operation from beginning to end, either successful or not.

Related thin-threads and test-scenarios can be grouped together to collectively describe all possible outcomes for an activity; for instance, (to use a well-worn example), when an ATM customer selects to withdraw money from a savings account there may be several outcomes. The obvious 'successful operation', where everything goes as planned with the customer withdrawing the desired amount from their own account; and then the other unsuccessful outcomes that may also occur; such as when our hypothetical customer either:

1. enters a wrong PIN number;
2. inserts the wrong card type;
3. selects the wrong account type; or
4. attempts to withdraw an amount larger than is available.

All of these scenarios, (and possibly others that are not included), make up an ATM *withdrawal* activity, or 'thin-thread group'. Each of these individual outcomes can be referred to as either a thin-thread or a US. For our purposes, we choose to refer to these outcomes as USs rather than thin-threads; a term we reserve for approaches focusing specifically on the development of test-cases.

[3] presents an approach to extracting thin-threads from Ads; and although their approach is quite exhaustive, it requires a considerable amount of manual processing in the conversion from an AD to an Activity Hyper-Graph and again in the collection of conditional expressions. The approach does not attempt to maintain concurrency and serializes the vertices discovered in Concurrent Regions (CR). There is currently no tool support for the approach.

[25] introduces a gray-box approach and tool (UMLTGF) for generating test-scenarios from ADs. Their approach, like that of [3] could not be fully automated. CRs are restricted to a maximum of two threads that have sequential 'CallActions' only in [25]. Interestingly, their algorithm does not appear to differentiate between branch nodes and fork nodes. Even with the 'two-thread restriction' we were unable to capture a pair of threads in a single scenario; thus, resulting in the capture of each thread as an alternate trace. This means that their approach may exclude many scenarios from a model, while at the same time producing incomplete traces.

[26] presents an approach that employs bacteria-like agents, optimizing an earlier algorithm [13] to extract thin-threads from ADs. This approach for generating test-scenarios directly from ADs is implemented in a tool called TSGAD (Test Scenarios Generator for Activity Diagrams). TSGAD lists the sets of test-cases derived from the artefacts between an AD's initial and final nodes in an output log file. These sets of test-cases can be clustered together to form a path through the AD artefacts. From these clustered paths it can be established which USs have been captured and which have not. They do not place the same restrictions on the structure of CRs and allow the inclusion of branches, merges and nested CRs.

IV. THE APPROACH

Although traversing the nodes and edges in a directed graph such as an AD, is not generally considered a difficult task – several algorithms perform this duty effectively such as DFS[1] and BFS[2]– when iteration and recursion are involved the difficulty is compounded. To ensure that a scenario does not traverse edges unnecessarily, the proposed method limits the number of times an edge can be traversed by applying a *status* variable to each AD artefact. A vertex's status can either be ACTIVE or INACTIVE; while an edge's status can be either FINISHED or UNFINISHED. The default status for a vertex is INACTIVE and an edge's default status is UNFINISHED.

During the processing of an AD for USs, each edge's source vertex has its status tested. If the current edge's source vertex is ACTIVE, then that vertex has already been encountered during the capture of the current scenario; therefore, the current edge's status is reset to FINISHED and it is processed once more as usual. This ensures that, outside of nested iterative loops, each edge is traversed at most twice.

When an edge with a FINISHED status is encountered, that edge is ignored and the process then moves to the next edge; or it begins to recurse until it finds an alternative route. This ensures that a branching edge (even an action node's self iterating edge) does not result in an endless loop or produce a scenario that is already represented with fewer iterations of the same trace-route. Each looping guard condition is tested in both the true and false conditions only; without performing boundary value analysis whereby, each potential expression value is tested at inside and just outside the expected limits for the situation.

When nested looping constructs are encountered there is the potential for scenario explosion and therefore, the number of times that an edge is traversed must again be limited. The situation depends on the number of outgoing edges that a nested branch node contains. Each internal edge must be traversed at most twice for each external branch's outgoing edge.

Recursion is a natural method of traversing directed graphs, but the problem becomes somewhat difficult when processing the nodes and constructs within a concurrent region. Particularly when one or more of the concurrent threads contains a looping or skipping construct. These constructs are identified by the sequence of the branch-merge node pairs. A

[1] Depth-First Search
[2] Breadth-First Search

branch followed by a merge signifies *skipping* while a merge-branch signifies *looping*. Section VI discusses these constructs in more detail.

V. THE ALGORITHM

Figure 1 is an AD depicting the algorithm employed by the proposed method. Once the artefacts have been captured by the preliminary functions, the *getNextAD(ADs)* method is called; which starts the usage-scenario (US) generation process. Any NULL artefacts are discarded. The function then determines the type of the received artefact. As each **Final** node (Z) in an AD is encountered, the process begins a new recursive trace by passing that node back to the *addToScenario(Artefact)* function. The *addToScenario(Artifact)* action first tests the validity of any artefacts that are passed to it. Each vertex's status is altered to ACTIVE when encountered. The received artefact type then determines which step is taken next. A vertex is passed to the *processVertex(V)* function; Edge artefacts are passed to the *processEdge(E)* function.

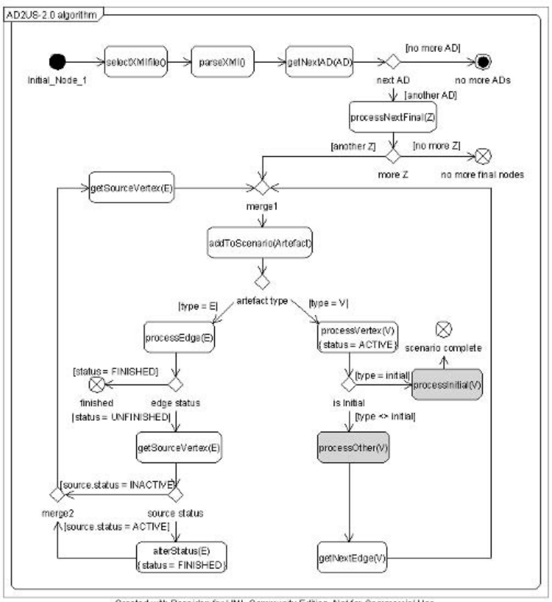


Figure 1 - An AD representing the AD2US algorithm

An edge's status is only altered to FINISHED when the status of its source vertex is ACTIVE. This indicates that the source vertex has been encountered earlier in the same scenario. If an edge's status is FINISHED, its source node is processed once more for the current US. This is necessary to ensure that each alternate edge is visited at most twice overall unless it is in a nested loop structure.

The algorithm is relatively straightforward and can be stepped through quite easily. The capture of usage scenarios in this manner works well, even in situations where models depict skipping, looping and complex concurrency like those described in section VI.

VI. DEALING WITH CONCURRENCY

A CR represents a set of threads, usually between a fork-join pair, that are logically processed in parallel. In a multi-processor or distributed system these concurrent threads can literally be processed simultaneously. Otherwise, processor time is shared by the threads of operation, which are performed sequentially until all threads are synchronized at a join node. This paper addresses single processor system models and therefore assumes that without edge weights; concurrent thread processing prioritization is determined by the system. The degree of complexity, specifically for CR modelling, has not been specified by the OMG; however, [26] provides a simple classification of basic CR modelling. Four categories are introduced to describe simple situations. [26]'s [3] categories are:

1. Atomic Fork Join;
2. Simple Fork Join;
3. Simple Nested Fork Join; and
4. Branch Nested Fork Join.

The problem is that these categories ignore the direction of processing with regard to looping and skipping. It is the opinion of the authors that these two factors must be included in a set of categories describing CR constructs.

Therefore, it is recommended that the following categories be considered:

1. Basic Fork Join (BFJ);
2. Looping Fork Join (LFJ);
3. Skipping Fork Join (SFJ);
4. Nested Fork Join (SFJ);
5. Skipping Nested Fork Join (SNFJ); and
6. Looping Nested Fork Join (LNFJ).

Figure 2 and Figure 3 depict CR segments for each of the listed categories. A BFJ is a CR that contains two or more threads; each of which contains only a sequence of action nodes and edges. Figure 2 shows three examples of separate CRs.

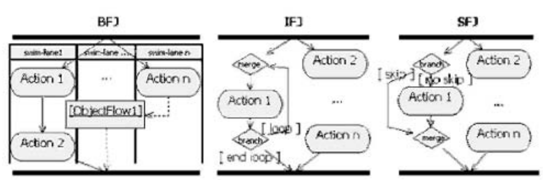


Figure 2 - examples of a basic fork-join CR, a looping and a skipping CR

The complexity that each of these constructs contributes to the processing of ADs, warrant their individual definition. In a recursive algorithm that processes ADs, a problem occurs when either a looping or skipping construct is nested in a CR.

[3] Page 386 - for more details.

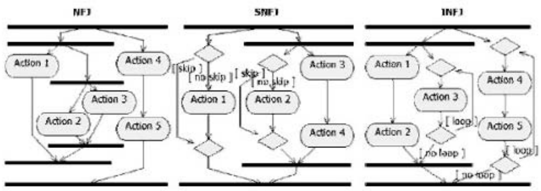


Figure 3 - examples of nested CRs; nested fork-joins, nested skipping and looping fork joins

The first example titled BFJ represents a Basic Fork-Join construct; it also includes swim-lanes which are used to define responsibility for actions. The title of the swim-lane typically names the actor involved in performing the action(s).

A LFJ is a CR which includes the same artefacts as a BFJ, in addition to at least one thread that includes a looping merge-branch node pair. A SFJ is a CR that contains a branch-merge node pair; which by their sequence indicates skipping.

Figure 3 depicts CR activity modelling constructs including nested CRs, and nested CRs with skipping and with looping. Like the previous examples the SNFJ and LNFJ activity segments, in Figure 3, do not depict real world activities; they are meant to depict the kind of sequences that are encountered within typical ADs. The SNFJ and LNFJ CRs illustrate constructs where skipping or looping are incorporated within both the outer and the inner CRs. Once more the thread configuration is not important; although it does present issues when scenarios are being generated automatically from constructs like these examples.

Regardless of whether the algorithm begins tracing scenarios at an initial node or at a final node, when processing reaches a branch within a CR, the algorithm must take into account each of the branch's alternate outgoing edges as a separate scenario.

This is either ignored or not taken into account in other approaches, [3], [25] and [26]. Each thread within a CR is traversed in some random order, without the processor knowing which thread may include a looping or skipping construct. For example, if the SFJ CR on the right of Figure 2 is being processed and the left thread is processed first, when the join node is reached and the processor recurses back toward the fork to process its next outgoing thread, it will encounter the alternate outgoing edge from the branch node and begin forward processing. This will result in both of the branch's outgoing edges being included in the same scenario.

To calculate the number of scenarios derived from a CR, we introduce the following: let $e^{(n)}$ represent the number of outgoing edges for a branch b_n, then in a CR that does not have nested branches, we can determine the number of possible scenarios derived from a CR with three branches as being:

$$\begin{aligned}\text{No. of Scenarios} &= b_1e^{(3)} \times b_2e^{(2)} \times b_3e^{(2)} \\ &= 3 \times 2 \times 2 \\ &= 12\end{aligned}$$

Where the first branch has three outgoing edges and the second and third branches each have two outgoing edges. Where a CR does have nested branches, the total number of nested-branch outgoing edges minus one, is added to the total number of outgoing edges of the non-nested branch. To clarify, if a second branch node is encountered before a merge node is encountered, then the second branch is considered to be nested within the first branch. Each successive nested branch has its total number of outgoing edges minus one, added to the nesting branch. For example:

$$\begin{aligned}\text{No. of Scenarios} &= (b_1e^{(3)} + b_2e^{(2-1)}) \times b_3e^{(2)} \\ &= (3 + 1) \times 2 \\ &= 8\end{aligned}$$

This process is repeated until the nesting is relinquished by the discovery of a merge node. The proposed method has successfully processed models containing constructs conforming to the suggested categories.

VII. The Case Study

The models which are applied to the compared approaches, contains eight ADs of varying complexity. Some of the ADs include non-real-world activities depicting CRs purely to expose the approaches to the categories discussed in section VI. Other ADs are included for their looping and/or skipping complexity. In addition, the model includes an AD with multiple initial nodes; and one of the models includes a diagram from OMG's UML 2.0 specification[4], this AD is a recognized example including many aspects of activity modelling; the Trouble-Ticket system. Obviously, room limitations prohibit the inclusion of all the ADs on which the methods are applied, however they are available in a technical report [8]. Each diagram used in this study contains one of the CR complexity categories based on those depicted in the examples in Figure 2 and or Figure 3.

A. The Comparative Criteria

The DoD guidebook states that one way to demonstrate test-case quality is to show full coverage; of the paths, conditions and data. The results produced by each of the compared methods are reconciled using a coverage criteria based on the DoD's E2E [18] report, specifically those of:

1. Path Coverage;
2. Condition (Branch) Coverage; and
3. Data Coverage.

In this case the coverage analysis is applied to the capture of USs based on a set of DoD E2E testing coverage criteria factors that are modified to reflect USs rather than thin-threads; these criteria are listed here:

1. percentage of Usage Scenario Groups (USG)s covered;
2. percentage of High-Risk Usage Scenarios (HRUS)s covered;
3. percentage of Usage Scenarios (US)s covered;
4. percentage of High-Risk Conditions (HRC)s covered;
5. percentage of Branch Conditions (BC)s covered; and
6. percentage of High-Usage Usage Scenarios (HUUS)s covered.

[4] page 292

USG refers to activity model diagram coverage. For the assessment of HRUS and HRC, we apply the DoD's usage-based test analysis. Whereby, scenarios are ranked according to their expected usage; their associated path conditions are then ranked according to the scenario to which they provide access. HUUS refers to the successful trace route through the activity. The results for processing each diagram are presented in Table 1 and Table 2.

More details of the results of processing these models and others are discussed in detail in the technical report as already mentioned [8].

B. *The Processing Results*

Each of the compared approaches is either walked-through manually, or has the model applied using the automated tool where available. Where manual operation is required each encountered artefact is recorded in a template as prescribed[5] by the DoD guidebook [18].

Table 1 results of correctly processing scenarios; showing the average across all diagrams

Method/ Author	% of USG	% of HRUS	% of US	% of HRC	% of BC	% of HUUS
[3] (m)	100	100	100	100	100	100
[25](m)	100	17.2	19.4	77.8	77.8	19.4
[26](a)	n/a	93.75	87.5	93.75	87.5	93.75
This approach	100	100	100	100	100	100

The figures in Table 2 express the number of USs captured by each approach for each of the diagrams in the case-study model. As the column headers suggest, each diagram contains specific artefact constructs representing the sequence of artefacts in that particular CR. The first six columns depict the specific pattern-like constructs listed in the suggested categories for CRs from section VI.

Table 2 results by number of USs generated per diagram

Method/ Author	BFJ	LFJ	SFJ	NFJ	LNFJ	SNFJ	MI	TTAD
[3]man	4	2	2	2	2	2	10	8
[25]man	21	3	3	5	4	4	10	16
[26]auto	4	0	2	2	0	2	5	8
This approach	4	2	2	2	2	2	10	8

Columns seven and eight in Table 2 are also included for their complexity. Column seven represents a diagram that depicts an AD that includes Multiple Initial (MI) nodes. Although both [3] and [25] make the assumption that diagrams do not include multiple initial nodes, in reality it is possible and legal AD modelling. An algorithm to effect this requirement is not difficult to implement and the manual approaches can process multiple initial nodes quite simply; therefore, we processed an example diagram manually using the algorithms of [3] and [25] and then attempted to process the same diagram with [26]'s automated tool but without success.

The final column in Table 2 depicts the results of each method when applied to the example OMG's Trouble Ticket AD. This diagram represents a trouble-ticket registration and service system activity from the UML 2.0 specification. The approaches listed in Table 2 are [3], [25] and [26] approaches; followed by the proposed approach described in this paper. Firstly, when the case-study model is processed manually and then when processed by the tool which implements the proposed approach.

We can compare the TSGAD approach and tool with the other approaches using the tool's test-scenario output. Once the TSGAD tool [26] has processed each diagram, it produces a log file of test-scenarios. The log file contains clusters of test-scenarios covering each possible path through the activity. When each cluster of test-scenarios is grouped together to form a thin-thread group, it can be seen that these groups also represent a US, from an initial node to a final node. From these groups, we can ascertain which paths through the activity have been covered and those that have not. The total number of USs covered by the approach for each diagram can then be added to Table 1.

The results in Table 1 indicate that [3]'s approach along with the proposed approach correctly produced the same number of USs from the activity model. However, [25]'s approach produced extra USs, (not variations of test-scenarios), from diagrams that include any form of CR. This is due to the approach's method of incorrectly counting each concurrent thread as a separate US. This becomes evident when stepping through the algorithm; as it does not differentiate between branch and fork nodes.

Xu's approach implemented by their tool produced a comprehensive set of test-scenarios for five of the eight diagrams. The TSGAD tool failed to generate any groups of test-scenarios from two diagrams; specifically, the diagrams that contain looping within CRs caused the tool to fail. The diagram with multiple initial nodes (MI) in Table 2 was tested using the TSGAD tool and again it failed completely; but when the second initial node was removed from the diagram, the tool successfully produced a comprehensive set of test-scenarios covering half of the USs.

C. *Implementation and Future Work*

The current version of the tool, which implements the approach, processes multiple ADs in a model and copes with more than one initial node in an AD. The tool has been developed to read XMI model files exported directly from Poseidon 2.4 and is currently being updated to read Poseidon 4.1 XMI files to comply with UML 2.0 and XMI 1.2. This means that the tool produces USs directly from a modelling application without manual intervention; representing a seamless progression from modelling tool to US without the risks of fault injection or incomplete scenarios sets. The tool has been built using Java (1.5) as a plug-in for the Eclipse environment.

Further to the case study used in this paper, a set of UML models are being developed that include more complex applications and diagrams that reflect real situations. These models along with an evolving technical report describing the models are available at [27]. The techniques and processes will be applied to the completed models to produce further evidence of the success of the approach.

[5] pages 18, 19

VIII. CONCLUSIONS

Although scenarios are becoming popular in both requirements engineering and test-case development aspects of system production, the automated capture of them has been elusive. Furthermore, there is a distinct lack of methods and tools to support the seamless progression from modelling tool to USs without the need for manual intervention. This paper has introduced an approach to their capture, beginning each trace-route at a final node and climbing back through the activity to an initial node to complete a US.

In addition, we have extended the CR description categories suggested by [26] to include skipping and looping constructs inside and outside of nested CRs. To compare several approaches for scenario capture, we have used example activity models that include diagrams with at least one of these categories or some other worthwhile feature. We have shown the results of manually stepping through the examples with each approach or applied the model using a tool that implements the approach.

The results are conclusive. [3]'s mostly manual approach accurately produces all scenarios as does the proposed approach; while the Gray-Box method [25] treats each CR thread as a separate scenario -- which is not correct -- and the TSGAD tool, although producing many clusters of test scenarios, which are grouped together to form USs, the TSGAD approach fails to capture all possible USs when looping is encountered inside CRs and more so in nested CRs.

Acknowledgement

Thanks to [26] for making the TSGAD tool available for comparison purposes.

References

[1] T. A. Alspaugh, D. J. Richardson, and T. A. Standish. Scenarios, state machines and purpose-driven testing. In *SCESM '05: Proceedings of the fourth international workshop on Scenarios and state machines: models, algorithms and tools*, pages 1–5, 2005.

[2] D. Amyot, X. He, Y. He, and D. Y. Cho. Generating scenarios from use case map specifications. In *3rd International Conference on Quality Software (QSIC'03)*, pages 108–115, Dallas, USA, 2003. IEEE Computer.

[3] X. Bai, C. P. Lam, and H. Li. An approach to generate the thin-threads from the UML diagrams. In *28th Annual International Computer Software and Applications Conference (COMPSAC'04)*, pages 546–552, Hong Kong, 2004.

[4] A. Bertolino, P. Inverardi, and H. Muccini. Formal methods in testing software architectures. In Formal Methods for Software Architectures: *Third International School on Formal Methods for the Design of Computer, Communication and Software Systems: Software Architectures, SFM 2003*, volume 2804/2003, pages 122–147, Bertinoro, Italy, 2003. LNCS.

[5] K. K. Breitman, J. C. S. d. P. Leite, and D. Berry. Supporting scenario evolution. *Requirements Engineering*, 10(2):112–131, May 2005.

[6] L. Briand, Y. Labiche, and G. Soccar. Automating impact analysis and regression test selection based on UML designs. In *International Conference on Software Maintenance (ICSM'02)*, pages 252–261, Montreal, Canada, 2002.

[7] R. Chandler, C. P. Lam, and H. Li. AD2US: An automated approach to generating usage scenarios from UML activity diagrams. In *12th Asia Pacific Software Engineering Conference (APSEC'05)*, pages 9–16, Taipei, Taiwan, December 15-17 2005. IEEE Computer Society.

[8] R. Chandler, H. Li, and C. P. Lam. Generating usage scenarios automatically from UML activity diagrams. Technical Report TR-06-01, Edith Cowan University, Western Australia, May 2006. Available: http://www.scis.ecu.edu.au/research/groups/se/selectedpublications.aspx.

[9] M. Gogolla, J. Bohling, and M. Richters. Validation of UML and OCL models by automatic snapshot generation. In *The Unified Modelling Language: Modelling Languages and Applications (UML'2003)*, volume 2863/2003, pages 265–279. LNCS, 2003.

[10] A. Hartman. AGEDIS: The final project report. Technical Report AGEDIS -1999 -20218, IBM, hartman@il.ibm.com , Feb 16 2004.

[11] G. Kosters, H.-W. Six, and M. Winter. Coupling use cases and class models as a means for validation and verification of requirements specification. *Requirements Engineering*, 6(1):14, 2001.

[12] W. J. Lee, S. D. Cha, and Y. R. Kwon. Integration and analysis of use cases using modular Petri nets in requirements engineering. *IEEE Transactions on Software Engineering*, 24(12), Nov 1998.

[13] H. Li and C. P. Lam. Using ant-like agents to generate test threads from UML diagrams. *LNCS*, SCI-E(Jan), 2005.

[14] X. Li, J. Hu, L. Bu, J. Zhao, and G. Zheng. Consistency checking of concurrent models for scenario-based specifications. *LNCS*, 3530(Jan):298–312, 2005.

[15] J. P. L´opez-Grao, J. Merseguer, and J. Campos. From UML activity diagrams to stochastic Petri nets: application to software performance engineering. In *WOSP '04: Proceedings of the fourth international workshop on Software and performance*, pages 25–36, 2004.

[16] B. Mitchell, R. Thomson, and P. Bristow. Scenario synthesis from imprecise requirements. *LNCS*, 3319(Jan):122–137, 2005.

[17] C. Nebut, F. Fleury, Y. Le Traon, and J.-M. J´ez´equel. Automatic test generation: A use case driven approach. *IEEE Transactions on Software Engineering*, 32(3):140–155, 2006.

[18] D. of Defense. End-to-end integration test guidebook. Technical report Version 1.0, USA Dept. of Defense, Dec 2000. Available: http://asusrl.eas.asu.edu/E2E/paper/dec28Book.pdf.

[19] J. Offutt, L. Shaoying, A. Abdurazik, and P. Ammann. Generating test data from state-based specifications. *The Journal of Software Testing, Verification and Reliability*, 13(1):25–53, 2003.

[20] OMG. Unified Modelling Language, v2.0 superstructure. Standard - Technical Report ptc/03-08-02, Object Management Group, April 2004 2004.

[21] G. Sabaliauskaite, F. Matsukawa, S. Kusomoto, and K. Inoue. An experimental comparison of checklist-based reading and perspective-based reading for UML design document inspection. In *2002 International Symposium on Empirical Software Engineering (ISESE'02)*, pages 148–160, Nara, Japan, 2002.

[22] J. E. Shin, A. G. Sutcliffe, and A. Gregoriades. Scenario advisor tool for requirements engineering. *Requirements Engineering*, 10(2):132–145, 2005.

[23] S. S. Som´e. Enhancement of a use cases based requirements engineering approach with scenarios. In *12th Asia Pacific Software Engineering Conference (APSEC'05)*, pages 25–32, Taipei, Taiwan, December 15-17 2005. IEEE Computer Society.

[24] A. G. Sutcliffe, N. A. Maiden, S. Minocha, and D. Manuel. Supporting scenario-based requirements engineering. *IEEE Transactions on Software Engineering*, 24(12):1072–1088, 1998.

[25] L. Wang, J. Yuan, X. Yu, J. Hu, X. Li, and G. Zheng. Generating test cases from UML activity diagram based on gray-box method. In *11th Asia-Pacific Software Engineering Conference (APSEC'04)*, pages 284–291, Busan, Korea, 2004.

[26] D. Xu, H. Li, and C. P. Lam. Using adaptive agents to automatically generate test scenarios from the UML activity diagrams. In *12th Asia Pacific Software Engineering Conference (APSEC'05)*, pages 385–392, Taipei, Taiwan, December 15-17 2005. IEEE Computer Society.

[27] R. Chandler, H. Li, and C. P. Lam. UML Models with Activity Diagrams: for Case Studies. Technical Report TR-06-02, Edith Cowan University, Western Australia, November 2006. Available: http://www.scis.ecu.edu.au/research/groups/se/selectedpublications.aspx.

Code Generation and Execution Framework for UML 2.0 Classes and State Machines

Romuald Pilitowski, Anna Derezińska
Institute of Computer Science, Warsaw University of Technology
15/19 Nowowiejska Street
Warsaw, 00-665 Poland

Abstract – **The paper presents Framework for eXecutable UML (FXU). FXU transforms UML models into programming code and supports execution of the resulting application according to the behavioral model. The code generation and execution is based on UML classes and their state machines. The FXU framework takes into account all concepts of state machines defined by the UML 2.0 specification. Ambiguities of UML state machine interpretation had to been resolved within the framework in order to obtain an executable application. During the runtime execution separate state machines and orthogonal regions are run as parallel threads. All kinds of events, states, pseudostates and activities are processed, as well. The framework was implemented and tested for C# code. The tool supports model-driven development of high quality applications.**

I. INTRODUCTION

Unified Modeling Language (UML) is a widely accepted, graphical language for specification, analysis and design of object-oriented computer systems [1]. Emergence of UML has triggered development of many tools and methodologies that improve quality of computer systems [2-9]. One of such methodologies is Executable UML (xUML) [10] that is intended to enable UML models to be translated into executable code. As such, an Executable UML model can be treated as Platform Independent Model (PIM) – the concept defined within the Model Driven Architecture (MDA). MDA is a recent initiative of Object Management Group consortium (OMG) which contributed to the development of the UML standard [2]. MDA is a model-centric approach to the software engineering based on models as major artifacts of the development process. However, none of the MDA tools supports complete and precise transformation of UML 2.0 state machines into executable C# code.

Our solution, called Framework for eXecutable UML (FXU) developed at the Warsaw University of Technology is an example of xUML tool. It is a code generator which enables transformation of UML 2.0 class diagrams and state machines into code in C# programming language, which can be further compiled and executed. A distinguishing feature of FXU is its ability to handle every single element of state machines as defined by the OMG standard – the UML 2.0 specification [1].

FXU consists of two components – FXU Generator and FXU Runtime Library. The FXU Generator transforms an UML 2.0 model into the corresponding code in the programming language. It takes into account information from class diagrams and state machines. The FXU Runtime Library is a library that implements entire logic of state machines as described by the UML 2.0 specification, including states, events, transitions etc. Both generated code and the FXU Runtime Library are required in order to enable execution of state machines.

The paper presents core algorithms that were implemented by the FXU Runtime Library. It supports execution of concurrent state machines which can specify behavior of many different objects. Regions of orthogonal states are executed concurrently as well. Transitions across vertices are triggered by events. The selection of transitions to be fired is based on their priorities. Queues of events are handled for each active state machine.

During development of FXU we faced problems of inconsistency of UML 2.0 specification concerning state machines. Some of the issues are semantic variation points. Semantic variation points are aspects that were intentionally not determined in order to leave its interpretation to a user. The order in which events are removed from the event pool is an example of the semantic variation point. However, there are also aspects such as the order of execution of multiple transitions, that were clearly overlooked in the specification. There are about ten inconsistency issues that cannot be resolved on the basis of the specification. All of them had to be identified and interpreted before the implementation of FXU. A detailed description exceeds however the scope of this paper [11].

Section II describes principles of code generation and the architecture of the FXU Generator. The FXU Runtime Library and its essential algorithms are presented in Section III. Related work is discussed in Section IV and final remarks end the paper.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 421–427.

II. Code Generation From UML Models

Code generators from UML models transform an abstract model of a software system into its implementation. Output of the most simple code generators from UML models [4,5,7] is only a stub of a system, e.g. only class declarations that have to be implemented. Approach of MDA is different. It assumes generation of the most components of the target system without much human intervention afterwards. FXU enables generation of fully functional applications as long as their structure and behavior can be represented using UML 2.0 class diagrams and state machines.

The distinguishing feature of FXU is the ability to generate and execute every single aspect of state machines described in the UML 2.0 specification. They are all listed in Tab. 1.

TABLE 1
Elements of State Machines Supported by FXU

States: simple states, composite (including orthogonal) states, entry-, do-, exit activities, submachine states.
Pseudostates: initial pseudostate, deep and shallow history, join, fork, junction, choice, entry and exit point, terminate.
Transitions: external, local, internal transitions, guards, actions, priorities of transitions.
Events: call events, time events, completion events, change events, signals, deferred evens, priorities of dispatching.

A. Code generation from class diagrams.

Generation from class diagrams to an object-oriented language is straight-forward. The most of the concepts from class diagrams have their counterparts in object-oriented languages. The FXU Generator and most other similar generators transform classes, their operations and attributes as well as interfaces into corresponding instructions in programming languages. Some generators including the FXU Generator support also generation of associations. The FXU Generator provides also with a template of generated code that can be edited in order to adjust outcome of generation to individual requirements.

B. Code generation from state machines.

Major goal of our solution is generation and execution of state machines. However, classes play an important role in both generation and execution of state machines because they create context for them. Every class can possess one or more state machines that describe behavior of its instances. Elements of the UML 2.0 model which are accessible within context of the class are also accessible within its state machines. Therefore, generation of state machines involves generation of corresponding classes.

Code generation from state machines is by far more difficult to accomplish than generation from class diagrams. The main reason for this is the lack of direct and precise mapping between concepts borrowed from state diagrams and general-purpose languages. For example, there is no counterpart of a transition or a state in the most general-purpose languages. Therefore every single concept defined in the UML 2.0 specification as a state, event, transition etc. was implemented and wrapped by a class of the FXU Runtime Library. The generated code consists mainly of instances of the FXU Library classes.

C. Code template

Figure 1 illustrates how FXU merges an UML 2.0 class and its state machines into one file with source code.

```
1  $visibility class $class_name {
2
3  $visibility [static] [readonly] $type $attribute_name;
4
5  StateMachine sm =  new
6          StateMachine($state_machine_name);
7
8  $visibility [static] $return_type $method_name
9  ($type $parameter_name1, $type $parameter_name2) {
10             $method_implementation
11 }
12 public void InitFxu(){
13         Create regions
14         Create states
15                 Add the states to the nesting regions
16                 Add activities to the states
17                 Add events deferred by the states
18         Create pseudostate
19                 Add the pseudostates to the regions
20         Create transitions
21                 Add triggers of the transitions
22                 Add guards to the transitions
23                 Add actions to the transitions
24 }
25 public void StartFxu(){
26         State::Enter ( sm );
27 }
28 }
```

Fig. 1. Template of generated source files.

Every class from the input model is generated as a separate file. Each class begins with declaration of its attributes (line 3). Keywords *static* and *readonly* are parameters of attributes and refer to UML properties of the same name. State machines of the class are treated as attributes of type *StateMachine* (rows 5-6). *StateMachine* and other types as *State*, *Transition*, *InitialPseudostate* are classes defined by the FXU Library.

Next, operations are declared (lines 8-9). If an operation is not abstract, its body is generated as well. Provided the input model contains an implementation, it is also included in the generated code. In case an operation triggers a call event, instructions broadcasting the appropriate event are added to the implementation of the operation.

If the class contains state machines, two methods *InitFxu* and *StartFxu* are also generated. *InitFxu* builds structures of state machines of the class. First, regions – sets of vertices that contain at most one active vertex, are created (line 13). Next, vertices, i.e. states and pseudostates are defined and added to the regions (rows 14-19). Finally, transitions across vertices are formed (20-23). The second method, *StartFxu* starts execution of all state machines owned by the class.

D. *FXU Generator*

The FXU Generator imports an XML Metadata Interchange (XMI) file with an UML 2.0 model. XMI is an OMG standard that can be used to exchange theoretically any model that can be expressed in Meta-Object Facility (MOF). Most present CASE tools support export of UML models to an XMI file, although several incompatible versions of XMI exist. We use an XMI variant supported by Eclipse sub-project called UML2 [12]. After import of the model, the FXU Generator applies transformation rules to class and state diagrams which results in generation of the corresponding source code. The source code may be obviously further developed by a user, but more importantly it can be compiled and linked against the FXU Runtime Library. The obtained executable binary file fully emulates the behavior of the input state machines.

Figure 2 depicts major components of the FXU Generator. It depends on two external libraries – UML2 and Jxp. UML2 is an open-source library for serialization and deserialization of UML 2.0 models stored in XMI format. UML2 library is utilized among others by Rational Software Modeler (version 6.0) and its more advanced edition Rational Software Architect [7]. Therefore, the FXU Generator integrates well with these CASE tools. Template processor Jxp is another external library used by the FXU Generator. We provide with Jxp templates for transformation of both classes and interfaces, which can be edited and customized by a user. The generator applies transformation rules to classes as well as state machines and uses Jxp for rendering the output files. Jxp combines the templates with the input model and generates the source code.

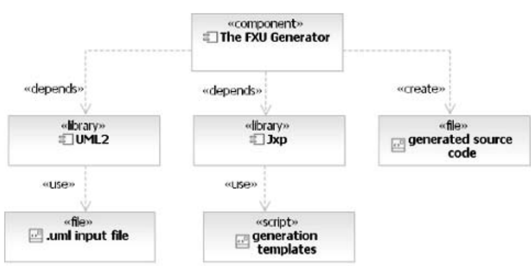


Fig. 2. Architecture of the FXU Generator.

III. FXU RUNTIME LIBRARY

A. *Overview of FXU Runtime Library*

The FXU Library implements the logic behind UML 2.0 state diagrams. A source code obtained as an output of the FXU Generator or written by hand is executed within the environment of the FXU Library. Figure 3 presents how a source code rendered by the FXU Generator and the FXU Library interact with each other.

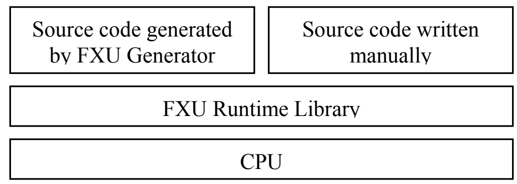


Fig. 3. Execution of state machines using FXU.

The FXU Library emulates behavior of UML 2.0 state diagrams. Because this behavior is universal for all UML 2.0 state machines, it was implemented as a component which is reused every time the state machines are being executed. The library consists of over forty classes. Each of them realizes a single concept defined by the specification such as a state, a transition, a fork pseudostate etc. Figure 4 depicts the essentials of state diagrams implemented in the FXU Runtime Library.

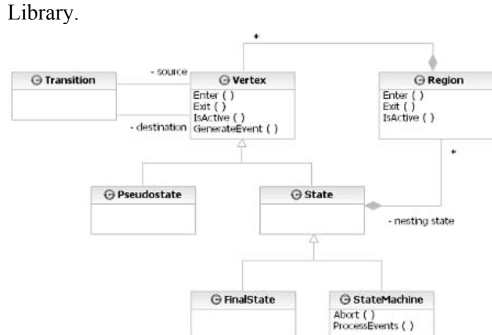


Fig. 4. FXU Runtime Library. Basic elements.

Execution of separate state machines and orthogonal regions is implemented through parallel threads. A state machine can react on different events defined in the UML model. The FXU Library realizes event processing (Fig. 5).

Each state machine has exactly one event pool which contains received but still not processed events. The order of events in the event pool is a semantic variation point. FXU treats events on the First-In First-Out basis. However, completion events have always the highest priority. It is the only exception enforced by the specification. An event can trigger any number of transitions. However, it has to be sent first. Events can be either broadcasted to all the state machines through *EventBroadcaster* singleton or can be sent directly to the selected state machines.

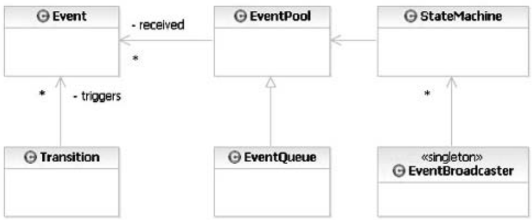


Fig. 5. FXU Runtime Library. Event processing.

B. *Algorithms*

State diagrams are by nature concurrent. Not only because more than one state machine may be running at the same time. Also a single state can contain more than one parallel region. Such a state is called orthogonal and also requires concurrent execution of some activities, e.g. traversing, entering substates or execution of *do activities*. Event processing is another task which works continuously for each active state machine and should be run simultaneously to the execution of a state machine. Every state machine disposes over own event pool which implements producers-consumer algorithm. Every element of a state machine can produce, i.e. send events but only the appropriate state machine can process them. For these reasons the FXU Runtime Library takes advantage of multithreading to a large extent.

We present core algorithms implemented by the FXU Runtime Library: the algorithm of execution of a state machine (Fig. 6), of entrance and of exit from a state (Fig. 7 and 8).

Execution of a state machine starts with its static validation (Fig. 6 line 3-4). Static validation verifies correctness of the structure of the state diagram, e.g. ensures that every initial pseudostate has exactly one outgoing transition. If the state machine is invalid, it cannot be launched.

Then, the state machine is entered (row 5). Detailed algorithm of entry to a state is presented in Figure 7.

Event processing and traversing is executed in the loop (lines 6-22). Its execution can be broken if either a final state of the state machine was entered or it was terminated. The state machine terminates when either a terminate pseudostate is entered or a critical malfunction in its behavior was detected, e.g. there are no enabled transitions outgoing from an active choice pseudostate.

In the next step, transitions to be fired are computed. First, the event with the highest priority is removed from the event pool (10-12). Then, currently enabled transitions are computed. An enabled transition is such a transition that is triggered by the event, whose source state is active and whose guard condition evaluates to true. The transitions to be fired are selected from the enabled ones. But because some transitions cannot be traversed at the same time, not every enabled transition will be fired. Priorities of transitions decide which of enabled transitions can be launched. However, the definition of transition priority provided by the UML 2.0 specification is not sufficient to resolve all conflicts. Therefore, we had to extend the definition of transitions priority. Selection of non conflicting transitions on the basis of our definition is possible in any state configuration. Its detailed description exceeds the scope of this paper [11].

Next, the deferred events are handled. If a deferred event fires no transition, it returns to the event pool (rows 15-19). A not-deferred event would be ignored and lost in such cases.

Finally, the selected non-conflicting transitions are fired in concurrent threads. In consequence no single transition is favored. After all transitions had been traversed, the next event can be removed from the event pool and the entire procedure repeats.

```
1  PROCEDURE
2          StateMachine::Enter(StateMachine: sm ) {
3  IF ( static validation of sm fails) THEN
4          RETURN
5  State::Enter(sm)
6  WHILE ( TRUE ) DO {
7  IF ( final state of sm was entered OR
8       sm was terminated ) THEN
9          RETURN
10 Let e be the event with the highest priority among all
11 events in the event pool
12         Remove event e from the event pool
13 Let M be a set of enabled transitions triggered by e
14 which were selected to be fired
15 IF ( set M is empty AND
16      exist active state which defers event e ) {
17         Add e to the event pool
18         CONTINUE
19 }
20 FOREACH transition t ∈ M DO
21         Traverse transition t in a concurrent thread
22 }
23 }
```

Fig. 6. Algorithm of execution of a state machine.

Figure 7 presents the algorithm of entry to a state. It applies also to state machines which are treated as the special, outermost states.

Before entrance to a state we assure that it is still inactive. Instructions in lines from 2nd to 4th are invoked as atomic, thread-safe operations.

Next, a new thread, called *after-thread* is created and fired asynchronously for each time event which triggers a transition outgoing from the state. The *after-thread* sleeps for the time interval defined by the corresponding time event, wakes up and adds it to the event pool (lines 5-8).

Then, all nesting states of the state **s** are entered recursively (line 10). It reflects the rule that nesting states are entered before the nested ones. Afterwards shallow and deep history of the nesting states are being updated (row 11 and 13).

After execution of *entry* activity (line 14), *do* activity of the sate is being executed in a separate, parallel thread, called *do-thread* (row 15). *Completion-thread* is another concurrent thread that sends its completion event for the state once the execution of *do* activity finished (line 16).

Next, the inactive regions nested by the state are activated. Activation of a region means entering an indicated vertex nested by this region or its initial pseudostate.

```
1  PROCEDURE State::Enter (State: s){
2  IF (state s is inactive) THEN
3          Mark state s as active
4  ELSE RETURN
5  FOREACH transition t outgoing from s DO
6          FOREACH event e triggering t DO
7                  IF( e is a time event) THEN
8                          Run after-thread
9  IF(exists state ns directly nesting s ) THEN {
10         State::Enter( ns )
11         Update shallow history of ns
12 }
13 Update deep history of all states nesting s
14 Execute entry activity of s
15 Run do-thread
16 Run completion-thread
17 FOREACH region r nested by s DO
18         IF ( r is inactive ) THEN
19                 Run region-activation- thread
20 }
```

Fig. 7. Algorithm of entry to a state.

Exiting a state (Fig. 8) is a process opposite to the entrance. First, we assure, that the state is still inactive (lines 2-4). If the state is inactive, it cannot be exited once again. Instructions in lines from 2nd to 4th are invoked as atomic operations and can be therefore safely executed by concurrent threads.

Then, *after-*, *completion-* and *do-thread* are stopped (rows 5-7). According to the UML 2.0 specification, *do* activity of a state may be terminated, if the exit had been requested.

After that, all the nested active states are being exited, which implements principle, that exit from nesting states follows exit from nested ones (lines 8-10).

Finally, the *exit* activity of the state is executed (line 12).

```
1  PROCEDURE Exit (State: s){
2  IF (state s is active) THEN
3          Mark state s as inactive
4  ELSE RETURN
5  Stop after-thread
6  Stop completion-thread
7  Stop do-thread
8  FOREACH region r nested by s DO {
9          Let ns be the only active state nested by r
10         State::Exit ( ns )
11 }
12 Execute exit activity of s
13 }
```

Fig. 8. Algorithm of exit from a state.

C. *Example of state machine execution*

The following example illustrates the process of state machine execution. The input model consists of a single class *TvSet* (Fig. 9) and a state diagram specifying its behavior (Fig. 10). The state machine has two states, a simple state *Off* and a complex state *On* with two orthogonal regions.

Fig. 9. TvSet class.

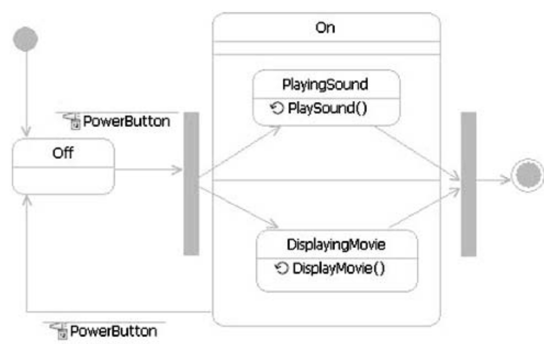


Fig. 10. State machine of TvSet class.

Provided that the above model is used as an input of the FXU Generator, a single file containing definition of the *TvSet* class and the implementation of its state machine would be the output of the generation. After compilation and linking against the FXU Runtime Library the state machine can be executed.

The execution of the state machine starts with entering its only initial pseudostate. Then, the state *Off* is entered. According to the algorithm (Fig. 7), entry-activity and then asynchronously do-activity are executed. After the do-activity finishes, a completion event for the state *Off* is sent. However, a call event *PowerButton* is the only event that triggers transitions outgoing from the state *Off* and the completion event is lost. If the *PowerButton* method is invoked now, the corresponding *PowerButton* call event is sent and transition to the fork pseudostate proceeds. Transitions from the fork pseudostate result in the execution of two concurrent threads that activate states *PlayingSound* and *DispalyingMovie* as well as their nesting state *On*. So long the state *On* is active, a transition back to the state *Off* can be fired, if the *PowerButton* event is raised again. Otherwise, after do-activities of states *PlayingSound* and *DispalyingMovie* finish, the corresponding completion events are sent. Consequently, transitions to the join pseudostate are entered and finally the final state is entered. As a result, the execution of the state machine is finished.

IV. Related work

FXU is one of the existing solutions that enable execution of state machines. Crane and Dingel have conducted survey of various different approaches to the problem of state machines modeling [3]. According to their study and our own research most of the code generators support only fundamental aspects of state machines described by the UML 2.0 specification. More advanced elements such as deferring events are generally not supported. A notable exception is Rhapsody of Telelogic AB (formerly I-Logix), which generates code from UML 2.0 state diagrams for C, C++, Ada and Java [6]. The unique feature of FXU is however its ability to transform *all* details of UML 2.0 state machines into C# code.

Two general approaches to code generation from state machines exist. The first one assumes no existence of any library or virtual machine which implement semantic of state machines [8, 13]. Therefore, large number of code must be generated even for simple state diagrams. The approach we prefer involves using a library (in our case the FXU Runtime Library) which hides all the logic of state diagrams [6, 9]. Therefore there is less code to generate. Consequently, generated sources are easier to understand and to edit. Such an approach enables also relatively easy reverse-engineering of state machines.

V. Final Remarks

The paper presented Framework for eXecutable UML (FXU). The framework was implemented in Java 1.5 (FXU Generator) and in C# 2.0 (FXU Runtime Library). The presented solution realizes all basic features of the class diagrams and the complete description of the UML 2.0 state machines. FXU was used to generate and execute C# code of over fifty UML models, including complex state machines and models with non-trivial class diagrams published on the Internet.

The purpose of FXU is to help developing robust applications in C#. The major advantage of FXU for a programmer is the efficient development of multithreaded, well-documented and maintainable applications.

FXU is the proof of the concept that complete and precise transformation of UML 2.0 state machines into executable code is possible. However, developing computer systems with FXU requires in-depth knowledge of state machines and is limited to specific class of problems, i.e. such problems that can be modeled with state diagrams.

Behavioral verification of UML models, especially state machines, can be provided using model checking techniques [14]. We are able to check whether UML state machines satisfy specified dynamic features, in the most cases analyzing the models translated into the appropriate notation of verification tools (e.g. SPIN[15]). Later, FXU can convert the same, verified UML models into an application, retaining the desired features after model to code transformation.

State machines are expected to attract more attention with increase of popularity of MDA tools which require more precise specification of computer systems.

References

[1] UML 2.0 specification, http://www.uml.org/#UML2.0
[2] The Object Management Group (OMG), http://www.omg.org/
[3] M. Crane, J. Dingel, *On the Semantics of UML State Machines: Categorization and Comparison.*, Tech. Rep. 2005-501, School of Computing, Queens University of Kingston, Ontario, Canada, 2005.
[4] Poseidon for UML, http://www.gentleware.com/
[5] MagicDraw UML, http://www.magicdraw.com/
[6] Rhapsody, http://www.ilogix.com/
[7] T. Quatrani, J. Palistrant, *Visual Modeling with IBM Rational Software Architect and UML*, 1st ed, IBM Press, May 2006.
[8] SmartState, http://www.smartstatestudio.com
[9] A. Knapp, S. Merz, C. Rauh. „Model Checking Timed UML State Machines and Collaborations", *in Proc. 7th Int. Symp. Formal Techniques in Real-Time and Fault Tolerant Syst.*, 2002, pp.395-414.
[10] S. J. Mellor, M. J. Balcer, *Executable UML a Fundation for Model-Driven Architecture*, Addison-Wesley, 2002.
[11] R. Pilitowski, *Generation of C# code from UML 2.0 class and state machine diagrams* (in Polish), Master thesis, Inst. of Computer Science, Warsaw University of Technology, Poland, 2006.
[12] F. Budinsky, D. Steinberg, E. Merks, R. Ellersick,T. Grose, *Eclipse Modeling Framework*, 1st ed, Addison-Wesley, Aug. 2003.
[13] Iftikhar Azim Niaz, Jiro Tanaka, "Mapping UML Statecharts into Java code", in *Proc. of the IASTED Int. Conf. Software Engineering*, Feb. 2004, pp. 111-116.

This page intentionally blank

[14] M. Gallardo, P. Merino, E. Pimentelis, "Debugging UML Designs with Model Checking", in *Journal of Object Technology*, vol. 1., no. 2, pp. 101-117, July-August 2002.

[15] G. J. Holzmann, *The SPIN Model Checker: Primer and Reference Manual*, Addison Wesley, 2003.

Laboratory instrumentation and object oriented design for working fluid control in an “absorption heat pump” using Water / Carrol

Rosenberg J. Romero[1,*], Antonio Rodríguez Martínez[1], Enrique Casillas González[2],
[1]Centro de Investigación en Ingeniería y Ciencias Aplicadas,
[2]Posgrado en Ingeniería y Ciencias Aplicadas
Universidad Autónoma del Estado de Morelos, Av. Universidad 1001, Col. Chamilpa, CP 62209, Cuernavaca, Mor. Mex.
* rosenberg@uaem.mx

***Abstract*— In this work the flow control in a single stage absorption heat pump (SSAHP) is shown, in particular case for the water – Carrol ™ absorption pair, and the operating conditions of the system are calculated. The dimensionless tendencies of powers are shown as a function of the final temperature of revaluation from waste energy. The used software is described for the temperature sensors calibration, as well as the software for in - line calculations of powers and mass flows into the thermodynamic cycle. Finally, a correlation for working fluid control into the Laboratory absorption heat pump is shown.**

Index Terms— absorption heat pump, thermodynamic cycle, operation conditions, water – Carrol™

I. Introduction

AN absorption heat pump of single stage (SSAHP) is a thermal device that is used in the industries for the recovery of waste heat for energy uses to higher temperature of the original source [1]. The operating of a SSAHP is based on heat and mass simultaneous transfer by absorption of a working fluid into an absorbent, with the consequent exothermic reaction that promotes increment of pressure and temperature in the fluids, this reaction is used to revalue thermal energy from a TEV temperature and recovered it at higher value: T_{AB}.

On the performance of a SSAHP, there are five basic components, two auxiliary systems and heat recovery system. This system has: vapor generator, a condenser, an evaporator, an absorber and an economizer [2], the auxiliary systems are a heating system and a cooling system [3] and the system of heat recovery consists of a heat exchanger coupled to the absorber, as schematized in the figure 1.

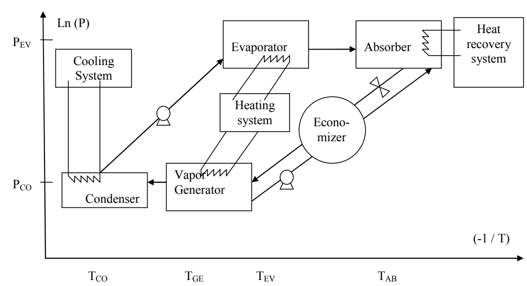


Figure 1. Schematic single stage absorption heat pump (SSAHP) for energy revaluation.

In a SSAHP, there are 32 variables to be able the operating of this cycle under steady state conditions. In the starting of this system, five critical parameters exist: Pressure, Temperature, Concentration of the absorbent, flow of working fluid and evaporation power.

The operating in a SSAHP had carried out in the "Ingeniería Térmica Aplicada" Laboratory (LITA) of CIIICAP – UAEM, in this operation we have been calculated and simulated in a personal computer with a previously validated model for the operation of a pilot plant [4].

The control that is carried out of the critical variables has been done in manual mode, with technical support to operate in permanent state. It is meritorious to mention that the starting of the system implies a transitory state of 5 hours for the control of the 32 variables. This transitory state was monitoring with a personal computer connected to a data logger with the software Benchlink©. [5]

II. Control challenges

In order to control the system, a variables correlation model

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 429–432.

has been used for the mixture water - Carrol. In this "working fluid - absorbent" (WFA) pair, the aqueous Carrol is the absorbent and the water (in phase vapor) is the working fluid.

With the used pattern [6] the operating conditions and the influence of the flow of the working fluid have been determined to fix the permanent state conditions.

We have determined the following methodology for the control of this pilot plant:

Modeling of absorption SSAHP system;
Simulation of water / Carrol as WFA pair;
Identification of Supporting with Benchlink © software
Developed data acquisition system with thermocouple,
Calibration for each made component. Some part of the calibration is shown in the figure 3.
Test of circuit with working fluid;
Object Oriented Design,
Test and validation of pattern – controller,
All this into the CIICAP – UAEM laboratory.

III. Results

Using the validated pattern, theoretical operation conditions has been determined to future real operating conditions in the pilot system at thermal applied engineering laboratory (LITA). These conditions are shown in the chart 1.

All conditions for the system was simulated with powers of 0.9 kW + / - 0.02 each component. With these conditions, it has been able to determine the ratio between the powers of generator and evaporator, as function of the revalorization temperature TAB. This behavior is shown in the figure 2.

T_{GE}	T_{CO}	T_{EV}	T_{AB}	P_{CO}	P_{EV}	X_{AB}	X_{GE}
ºC	ºC	ºC	ºC	kPa	kPa	%w	%w
69.9	16.1	71.3	90.0	1.83	32.9	49.1	73.8
69.9	16.1	71.3	91.0	1.83	32.9	50.0	73.8
69.9	16.1	71.3	92.0	1.83	32.9	50.9	73.8
69.9	16.1	71.3	93.0	1.83	32.9	51.7	73.8
69.9	16.1	71.3	94.0	1.83	32.9	52.4	73.8
69.9	16.1	71.3	95.0	1.83	32.9	53.1	73.8
69.9	16.1	71.3	96.0	1.83	32.9	53.8	73.8
69.9	16.1	71.3	97.0	1.83	32.9	54.5	73.8
69.9	16.1	71.3	98.0	1.83	32.9	55.1	73.8
69.9	16.1	71.3	99.0	1.83	32.9	55.7	73.8
69.9	16.1	71.3	100.0	1.83	32.9	56.3	73.8
69.9	16.1	71.3	101.0	1.83	32.9	56.9	73.8
69.9	16.1	71.3	102.0	1.83	32.9	57.5	73.8
69.9	16.1	71.3	103.0	1.83	32.9	58.0	73.8
69.9	16.1	71.3	104.0	1.83	32.9	58.6	73.8
69.9	16.1	71.3	105.0	1.83	32.9	59.1	73.8
69.9	16.1	71.3	106.0	1.83	32.9	59.6	73.8
69.9	16.1	71.3	107.0	1.83	32.9	60.2	73.8
69.9	16.1	71.3	108.0	1.83	32.9	60.7	73.8
69.9	16.1	71.3	109.0	1.83	32.9	61.2	73.8
69.9	16.1	71.3	110.0	1.83	32.9	61.7	73.8

Chart 1. Simulated conditions for pilot system in LITA.

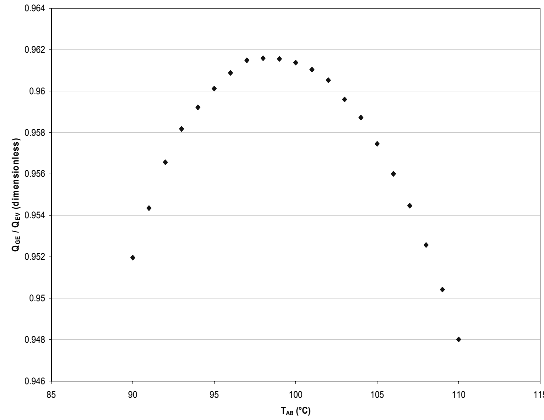


Figure 2. Ratio of powers in generator and evaporator as a function of revalorization temperature.

IV. Data Acquisition

Supporting with Benchlink © software we have developed data acquisition system with thermocouple, whose previous calibration for each component has made. Some part of the calibration is shown in the figure 3.

V. Flow control

To carry out the flow control toward the system we have programmed the calculation of the powers of the components in the HP-VEE © of Agilent Co. software. The calculation has been complicated because it requires to be carried out readings in real time for each temperature, to manipulate the data involved in each component to be able the calculation by energy balance considering thermodynamic first law, to define thermal load in each component.

Some part of the program is shown in the figure 4. This program has prepared with a different color each component in order to avoid the interaction of mistaken thermocouples.

Channel			Measurement	Scaling (Mx+B)				Comment
ID	Scan	Name	Function	Scale	Gain(M)	Offset(B)	Label	
201	☑	Sol HEX LBr out gen	Temp (type T)	☑	0.9986	0.0634	C	LiBr que sale del generador y entra al SOL HEX
202	☑	HEX2 H2O out c-H e-C	Temp (type T)	☑	0.998	0.2691	C	(cond) H20 pura caliente (enf) H20 pura fria que sale del HEX2
203	☑	HEX2 -HW in	Temp (type T)	☑	0.9866	1.1342	C	Agua del medio de enfriamiento que entra al HEX2
204	☑	Sol HEX LBr out abs	Temp (type T)	☑	0.9986	0.0127	C	LiBr que sale del absorbedor y entra al SOL HEX
205	☑	HEX2 W out c-H e-C	Temp (type T)	☑	0.9995	0.121	C	(cond) Agua del medio de enf (enf) agua de enf que sale del HEX2
206	☑	HEX2 H2O v in	Temp (type T)	☑	1.0008	0.1652	C	H20 vapor que entra al COND AUX o al HEX2 (cond)
207	☑	Sol HEX LBr in abs	Temp (type T)	☑	0.9994	0.1031	C	LiBr que sale del SOL HEX hacia el absorbedor
208	☑	Cond Aux -HW out	Temp (type T)	☑	0.9996	0.0964	C	Agua del medio de enfriamiento que sale del COND AUX
209	☑	Generador LiBr out	Temp (type T)	☑	0.9994	0.1328	C	LiBr concentrado que sale del GENERADOR
210	☑	Cond Aux H2O out	Temp (type T)	☑	1.0117	0.2205	C	H20 pura caliente que sale del COND AUX
211	☑	Cond Aux -HW in	Temp (type T)	☑	0.9987	0.2907	C	Agua del medio de enfriamiento que entra al COND AUX
212	☑	HEX1 -HW in	Temp (type T)	☑	0.9994	0.0598	C	Agua del sistema de calentamiento que entra al HEX1
213	☐		DC volts	☐	1.0	0.0	VDC	
214	☐		DC volts	☐	1.0	0.0	VDC	
215	☐		DC volts	☐	1.0	0.0	VDC	
216	☐		DC volts	☐	1.0	0.0	VDC	
217	☐		DC volts	☐	1.0	0.0	VDC	
218	☐		DC volts	☐	1.0	0.0	VDC	
219	☐		DC volts	☐	1.0	0.0	VDC	
220	☑	Presión Barométrica	DC volts	☑	1.0	609.74	mmHg	Presión barométrica
221	☐		DC current	☐	1.0	0.0	ADC	
222	☐		DC current	☐	1.0	0.0	ADC	

Figure 3. Gain and offset for thermocouples in the system.

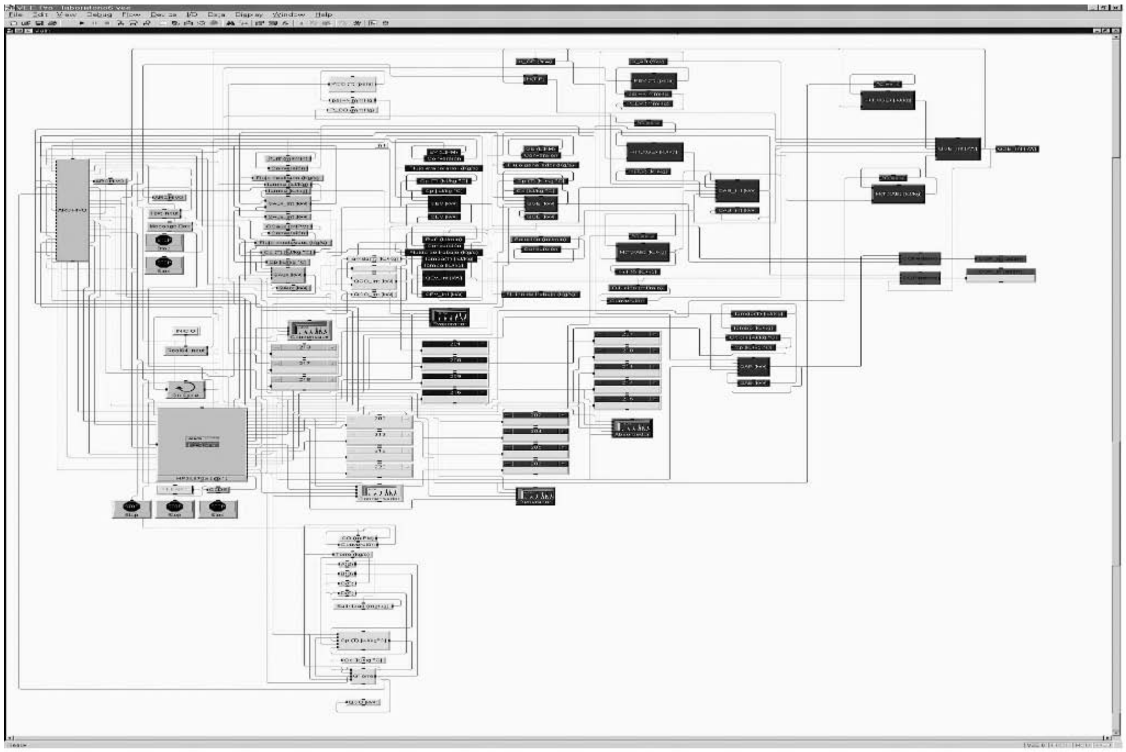


Figure 4. Object oriented design for powers calculation in the SSAHP.

I. WORKING FLUID CONTROL

In this first stage of this project, we have been able to carry out the modeling of the SSAHP with the WFA water / Carrol, we gauge temperature sensors and we have been able to manipulate the data in real time to be able the calculation of the components powers. The foreseen flow control system for working fluid in this system will be carried out with the manipulation of the computer file generated in real time for Benchlink and coupled to the printer port by HP-VEE for the control of working fluid pump frequency [7]. Whit this value, we calculate that working fluid flow has the ratio for thermodynamic first law of absorbent concentration: then in the absorber mass balance, the ratio of flow was calculated as

$$M_{EV} = M_{AB} - M_{GE}, \tag{1}$$

Where M_{EV} is the working fluid mass flow from the evaporator to the absorber, M_{AB} is the working fluid plus absorbent from absorber to generator, and M_{GE} is the working fluid plus absorbent going from generator to absorber.

With mass absorbent balance, we conclude than without absorbent into the evaporator, the product X_{EV} M_{EV} goes to a null value [4]:

$$0 = X_{AB} M_{AB} - X_{GE} M_{GE}, \tag{2}$$

where X_{AB} is the ratio of absorbent and the total mass into the absorber, X_{GE} is the ratio between absorbent and total mass into the generator, X_{GE} is always higher to X_{AB},

Therefore

$$X_{AB} M_{AB} = X_{GE} M_{GE}, \tag{3}$$

$$\text{Hence: } X_{AB} / X_{GE} = M_{GE} / M_{AB}, \tag{4}$$

Using the data acquisition of data logger, the energy power of absorber and generator was calculated, and then the mass flows M_{GE} and M_{AB} are known, so the working fluid flow M_{EV} is calculated. This is a second order behavior for exothermic heat dilution from X_{GE} to X_{AB}.

Now, using the ratio of Q_{GE} / Q_{EV} as a T_{AB} function, for each calculated Q_{GE}, there is only a Q_{EV}, which for thermodynamic first law depends on working fluid flow, only.

II. Conclusions

The Laboratory absorption heat pump of a single stage (SSAHP) with the mixture water / Carrol ™ has been simulated and instrumented. A maximum power ratio between generator and evaporator has defined. The calibration of the thermocouples leads that gains are 1 and the offsets spread to 0. In Laboratory control of the flow we need of the manipulation of the in line data, for that the acquisition software was not enough for this objective, so change to another software that allows the manipulation. Finally, the control is subject to the ratio of power in the evaporator and we have calculated the theoretical non linear function for its implementation.

III. Acknowledgment

To CONACYT SNI 52028 (2007), UAEM – CA70 projects and, to CIICAP-UAEM for its installation support.

IV. References

[1] Bourouis M., Coronas A., Romero R. J., Siqueiros J., "Purification of seawater using Absorption Heat Transformers with water – (LiBr+LiI+LiNO3+LiCl) and low temperature Heat Source", Desalination, Volume 166, Pp 209 – 214, 2004

[2] Romero R. J., Rivera W. and Best R., "Comparison of the theoretical performance of a Solar Air Conditioning System operating with water / lithium bromide and an aqueous Ternary Hydroxide", Solar Energy Materials And Solar Cells, V 63, Pp 387 – 399, 2000

[3] Cerezo Román Jesús, "Simulación de transformadores de calor acoplados a procesos industriales y evaluación experimental de un transformador térmico de 2 kW de potencia", Tesis de maestría, UNAM, México, 2002

[4] Rivera W., Romero R. J., Best R. and Heard C. L., "Experimental evaluation of a single stage heat transformer operating with the Water/Carrol mixture", Energy The International Journal, Vol. 24 (4), pp 317 - 326, 1999

[5] Romero Rosenberg J., Barajas Mendoza Aris, Rayón López Liliana; "Instrumentación y Control de flujo refrigerante en una bomba de calor por absorción de una etapa con la mezcla bromuro de litio – agua", Memorias del XXVII Encuentro Nacional de la AMIDIQ, Ixtapa-Zihuatanejo, Guerrero 2 al 5 de Mayo, Pp. PR054, Mexico, 2006

[6] Romero R. J., Rivera W., Pilatowsky I. and Best R., "Comparison Of The Modeling Of A Solar Absorption System For Simultaneous Cooling And Heating Operating With An Aqueous Ternary Hydroxide And With Water/Lithium Bromide", Solar Energy Materials And Solar Cells, V 70, Pp 301 – 308, 2001

[7] Alanís Navarro José Andrés, Labastida Del Toro Darío Carlos, Fernández Resenos Maribel, Romero Rosenberg J., "Control de flujo refrigerante en una bomba de calor por absorción de una etapa", Memorias del encuentro de Investigación en Ingeniería Eléctrica, Zacatecas, Zac., Abril 5 — 7, 2006

Dr Rosenberg J Romero D. (B'93, MS'96, PhD'01) Main author is a Chemical Engineer for the Puebla's Autonomous University, Master in Photothermal Sciences and Engineering PhD for the Mexico's Autonomous National University.

This young doctor is working for the Morelos' State Autonomous University, he is a researcher professor in "absorption heat pumps" since 1996, and has several international papers with relative themes. Actual efforts of him are the automation, higher heat & mass transfer effectiveness and applied thermodynamic studies.
Dr. Romero has been nominated and granted this year, with the "Alfonso Caso" medal, for the higher and significantly innovation in the Mexico's Autonomous National University studies.

Dr Antonio Rodriguez Martinez. (B'93, MS'02, PhD'05) Secondary author is a Chemical Engineer for the Morelo's Autonomous University, Master in Chemical Sciences and Process Engineering PhD for the Rovira I Virgili University.

This young doctor is working for the Morelos' State Autonomous University, he is a professor in chemical engineer since 1995, and has several international papers with relative themes. Actual efforts of him are the computer simulation, process automation and applied sciences studies.
Dr. Rodriguez has been nominated and granted two years, with the "Best project PhD" and "Feasiblest project", for the innovation in the Generalitat de Catalunya, Spain.

I.Q Enrique Casillas González. (B'05) Third author is a Chemical Engineer for the Morelos' Autonomous University, he is working in his Master degree in Engineering and Applied Sciences for the Moreslos' Autonomous University.

This young engineer is a researcher in absorption heat pumps since 2004, and has national papers of this theme. His actual studies are in the automation of applied thermodynamic cycles.
I.Q. Casillas has been granted lst year, with the "Honorific mention", for the higher efforts in the Bachelor Morelos' Autonomous University studies.

Towards the Use of Mediated Knowledge-based and User-defined Views in Super-peer P2P Systems

R. Mohamed & C.D.Buckingham
School of Engineering & Applied Science
Aston University
B4 7ET, UK

M.B.Al-Mourad
School of Computing & IT
University of Wolverhampton
WV1 1SB, UK

Yaser M. A. Khalifa
Electrical & Computer Engineering,
State University of New York

***Abstract* - In recent years, peer-to-peer data integration systems have attracted significant attention for their ability to communicate, collaborate and share information in a networked environment. One of the main problems that arises in such systems is how to exploit their mappings in order to answer queries posed to one peer. Our proposed framework can be used to exploit the existing mapped data together with its data location information for defining a peer's data view. This data view is expected to produce query results based on peer preferences rather than using standard query processing at the super-peer level, as practiced in current super-peer P2P systems. Our framework consists of two major components: a mediated knowledge-base at the super-peer and user-defined data views at the peer.**

I. INTRODUCTION

Recently, Peer-to-Peer (P2P) systems have become an active research area because of the opportunities for real-time communication, ad-hoc collaboration and information sharing in a large-scale environment. P2P refers to a class of systems and applications that employ distributed and autonomous nodes (peers) that co-operate in the community to share resources and services [1,2]. As the actual data is stored in various autonomous peers' data source locations, peers are 'linked' to other peers by mappings. Two basic problems arising in this architecture are: how to discover, express and compose the mappings between peers and how to exploit the mappings in order to answer queries posed to one peer [3]. The second problem is studied in this paper.

In P2P systems, peers are connected on an ad-hoc basis and the location of information is not controlled by the system. There is no guarantee that a query will be successful, even for the best query language. This is because peers only have local knowledge of the network, within which peer nodes may enter and leave frequently. The most widely known P2P network architectures are pure and super-peer networks. Our work will investigate searching issues in super-peer networks.

A search process includes aspects such as the query forwarding method, the set of nodes that receive a query-related message, the form of these messages, local processing, the stored index, and information maintenance [4]. The process depends on the system architecture used. Basically, searching can be classified as blind search, and informed search. Blind search is formally used in pure P2P systems. The original Gnutella [5] algorithm used the blind search method (also known as a flooding scheme) where each posted query is forwarded to all accessible peers within TTL ('Time-To-Leave') hops [4].

Super-peer networks use the informed search method because of the function of the super-peer node [6, 7, 8, 9], which can be seen as a hub that receives queries from connected peers (leaves). This query is forwarded to any relevant leaves and also to neighbouring hubs based on a query routing table (also known as a super-peer index). The super-peer index retains information about data stored at neighbouring peers so that the posted query will be forwarded to relevant peers only, in order to reduce the network traffic. This index would maintain either the actual location of required data as in Gnutella2 [10] or give 'directions' towards required data as in 'Routing Indices' (RIs) [11]. Hence, the super-peer node becomes the most vital component for query processing in super-peer networks. However, super-peers that provide the index for query routing are burdened by other peer nodes and by having to transform posted queries into sub-queries that become local queries for peers [23]. This project aims to reduce this burden by providing individual peers with the capability to define their own data views so that they can process posted queries locally, based on peer preferences (peer preference refers to the particular data required by a peer, where queries can be filtered and constraints imposed to return data suited to the requesting peer).

In this paper, we intend to address the aforementioned problems by proposing mediated knowledge-based and user-defined views in super-peer networks. The mediated knowledge-base aims at linking the routing index table to information about data residing at the peer locations. This knowledge base will then be used by peers to save queries made to the super-peer so that when a peer is given the same query again, it can execute the query directly without having to go through the super-peer. These saved queries are known as user-defined views and will function as a local search mechanism in order to produce query results with respect to peer preference.

The paper is organised as follows: Section 2 introduces a scenario in tourism which is used as a practical paradigm for further study. Section 3 reviews P2P data integration systems based on super-peer networks, with some motivating issues for our project. Section 4 and the

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 433–438.

subsequent sub-sections present our framework. Conclusions and a comparison of our proposed framework with centralised web-server and existing super-peer P2P systems are in Section 5.

II. SCENARIO

Consider a tourism scenario that consists of a Customer, a Basic Service (BS), and an Additional Service (AS) as shown in Figure 1. The BS and AS are information providers to the web portal. Companies who offer simple transportation services or accommodation, such as airlines and hotels, are consider as BS, while tourism and tour operator services are considered as AS. Customers require travel information from both BS and AS, while AS needs information from BS, for example, to generate travel packages. From the information provider's perspective, each of BS and AS should publicise their services on the World Wide Web (WWW). For example, hotels need to publish seasonal price information to attract more customers. This information is required by customers and also AS. At the same time, a travel agency (part of AS) may include this hotel promotion as part of its travel promotion packages that are also needed by the customer. However, this kind of information exchange that is currently available in centralized web servers among different BS and AS services is time consuming to obtain, out-of-date, and error prone, even though it is often available electronically at every level. Furthermore, different organisations may have different database schema to capture their data.

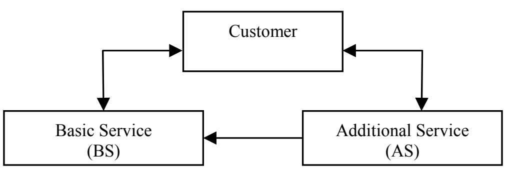


Fig.1. Relationships between three roles in a tourism scenario.

With existing WWW search engines, if customers want travel information about 'Kuala Lumpur' in Malaysia, they may type 'Kuala Lumpur travel agency' into the WWW search engine Some results containing the keywords return but may not contain any web pages for travel agency services at other areas in Malaysia because the exact keyword of other locations has not been specified.

Compared to centralized web servers, P2P systems have a distinct advantage as information providers because peers have more autonomy, both in providing their data to be shared with the community in the web portal and by maintaining their personal data. Additionally, end users can directly establish connections with other users (peers) without involving the centralized web server. Hence, we propose a framework for producing query results that considers peer preferences through user-defined data views. In this framework, the super-peer node is assisted with a knowledge-base that contains the actual location for particular information. Any peer could request these information locations to define their own data views (i.e. generate queries to other peers directly, without having to engage the super-peer). As illustrated in Figure 2, one of the peers is also a web portal and the AS information provider should be able to capture data required by the web user. An added benefit of using a knowledge-base for maintaining locations of information is that more intelligent partial keyword matching can be accomplished. The idea is that the knowledge base will generate more flexible queries that, along with the location information, can be stored as local data views within peers and executed in future without depending on super-peer query processing.

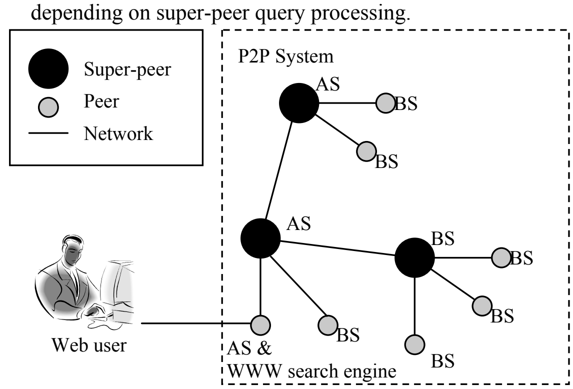


Fig.2. Illustration of a scenario in a super-peer P2P system network.

III. REVIEWS OF P2P DATA INTEGRATION SYSTEMS

P2P data integration systems are networks of autonomous peers that have recently attracted significant attention as an effective architecture for decentralized data sharing, integration and querying. Each peer shares a part or all of their resources with the community. In general, the success of such systems is achieved by increasing numbers of participants, thereby incrementing data storage and computational power of the whole system. However, as pointed out by Gribble et. al. [12], often generic P2P systems do not properly manage the semantics of data exchange. This situation leads to some drawbacks about availability and consistency of the service provided by a P2P system. It happens because of the absence of a global information schema or global knowledge for the whole peers' community. We will review some P2P data integration systems that try to avoid this by being based on a super-peer network. We are interested in how information can be shared among peers to help produce query results.

In the Edutella project [8], each participating peer has to obtain an RDF schema to be shared in the community. Additionally, each vocabulary used in the schema has to be based on a shared vocabulary dictionary. These limitations lead to decreased peer autonomy. In contrast, peers in the ORCHESTRA system [13] are free to publish and use any schema (from the same domain) for sharing in a collaborative data sharing system. There is no limit on the number of peers that can simultaneously publish and reconcile their actual schema to be shared. Peers are partially ranked by authority to resolve any schema reconciliation conflicts automatically. Yet, there is no solution for schema conflict between the same ranking peers. On the other hand, AutoMed system [6] provides a pre-defined global schema at the super-peer node. Each participating node would map their local schema through a 'transformation pathway' and any queries would be processed using this transformation. However, this flexible query processing gives some drawbacks to the system, where automating query reformulation has exponential time complexity with respect to the number of query and source schema. In Piazza [7], data is shared among pre-selected peers using composition mapping that works by optimizing the query processing in P2P. The composition mapping process is done by the super-peer node but it has no ability to semi-automate the specification of source descriptions, because the authors claim that composition mapping cannot be done at run-time.

P2P data integration systems allow autonomous peers to share data by the super-peer node acting as a mediator for the data. However the shared data are mutually inconsistent because each peer's own internally consistent data instances may conflict with others. Therefore, a mediated schema at the super-peer has to be maintained in order to reconcile the inconsistency and provide a consistent query answer [14].

Maintaining global semantic consistency in a distributed environment leads to the un-decidability of query answering [15]. Hence, the "What-to-Ask" (WTA) approach is proposed in [16], using an ontology-based framework. In WTA, a query-answering service is available at the queried peer but any changes in the representation formalism of WTA may seriously affect the ability of dealing with information exchange. Therefore, the knowledge that is being used by interoperating peers cannot be updated. On the other hand, a concept of 'peer agreement' for global schema repair has been introduced in [17]. It aims to have a peer preference that influences the process of repairing the global schema. Thus a query result is produced based on a repaired global schema by considering the peer preference. Unfortunately, checking whether all peers are satisfied with the shared schema is a very ambiguous process, and a difficult task that is unlikely to be feasible in polynomial time.

In conclusion, there are several issues that need to be highlighted from the aforementioned drawbacks. Firstly, autonomy of super-peer nodes should be considered, where leaf nodes are able to manage query processing by themselves. Secondly, the shared knowledge should be easy to maintain, with updating done without affecting the existing system's ability. Last but not least, peer preferences should be considered when answering the queries. A practical approach towards achieving these aims needs to be considered.

IV. PROPOSED FRAMEWORK

Based on the scenario in Section 2, an example P2P system framework is proposed in this section. A general P2P system in our project is defined as $P = (\mathcal{P}, I, \mathcal{N})$, where:

i. $\mathcal{P}$ is a non empty set of individual peers denoted as p. Each peer $p \in \mathcal{P}$, where $\mathcal{P} = (p_1, p_2, \ldots, p_n)$.

ii. I is the data integration between source schema at p and global schema at the super-peer node. The integration of p is formalised as $I(p) = <\mathcal{K}, \mathcal{S}_p, \mathcal{M}_p>$, which is the knowledge ($\mathcal{K}$) of P, source schema ($\mathcal{S}$) of p and mapping ($\mathcal{M}$) of p.

iii. $\mathcal{N}$ is a neighbourhood function that provides a set of peers $\mathcal{N}(p) \subset \mathcal{P} - p$.

In this project, schema mapping at the data source level is not being considered, since we assume that the schema used at a source is ready to be mapped to the global schema. Specifically, we assume that all peers can be mapped to an ontology at the super-peer, where ontology is defined as a term used to refer to the shared understanding of some domain of interest [18]. In [19], an ontology is used to present complex concepts in terms of a pre-defined and limited vocabulary. We define ontology as the number of corresponding pre-defined classifications used in P: $C(P) = \mathcal{K} = (c_1, c_2, \ldots, c_n)$. In effect, $\mathcal{K}$ is an instantiation of these classifications to produce the system ontology.

The following sub-section discusses the mediated knowledge-based view, proposed for supplementing the super-peer routing index table to facilitate grouping of data source information among leaf nodes. It is followed by user-defined view structures that aim to forward queries to the right sources as a local query.

A. Mediated Knowledge-base

The knowledge-base in super-peers performs as a dynamic mediator between peers, where data-source information from participating peers can be added or removed without re-structuring the shared schema integration. The data source information is required to facilitate the query routing table in order to forward the query or sub-query to an exact peer location as a local query. The basic idea of maintaining the data source information is adopted from [20]. In [20], the pre-defined shared schema is presented as an XML schema structure, whereas our project maintains the shared schema as an inter-related class of knowledge in ontological form. The reason for using an ontology is that classes (in our case a shared schema) can have relationships that are more flexible compared to the hierarchical approach of XML. Thus, the relationship between schemas can be easily identified. This relationship is essential for the searching process, especially when the keyword used is not exactly matched. In addition, ontology classes and relationships are possible to update.

A schema in Figure 3 is a class or subclass, which is represented by the "Classes of $\mathcal{K}$" column in TABLE 1. The attributes of each class are shown in the second table column and subsequent columns illustrate the mappings between peers and the ontology. Each schema from a peer,

p, is assigned to exactly one class, c, from the list of shared schema in $\mathcal{K}$. In the mapping process, a peer must provide the exact location of data source information for $c(p)$ to be matched with a particular $c(\mathcal{K})$. Then, peers have to define a mapping function, mf, for each mapping attribute, $a(p)$, that is not exactly matched with $a(\mathcal{K})$. The mapping function is useful for generating a local query. For example, the attribute 'price' of class $\mathcal{K}$ may use dollars while peer p_1 might have its attribute 'cost' based on pounds. Therefore, p_1 needs to prepare a currency converter from pounds to dollars as a mapping function when assigning 'cost' to 'price'. This is shown in TABLE 1 by the last row entry in the column for p_1. The price attribute for the hotel class, $\mathcal{K}$, corresponds to the cost attribute of p_1 and there is an associated mapping function to convert from pounds to dollars.

The abovementioned example is a one-to-one mapping but there may be one-to-many and many-to-one mappings. For one-to-many mappings, attributes such as 'address 1', 'address 2', 'postcode', and 'city' of p_1 must be assigned to a single 'address' attribute in $\mathcal{K}$, which p_1 achieves using a union function on the separate attributes to produce one 'address'. In the case of many-to-one mappings, such as between 'country-code' and 'phone-number', a subset function will identify the first three digits to be the 'country code' and the rest is assigned to 'telephone number' as a mapping function for p_2. TABLE 1 illustrates the reconciliation of $C(\mathcal{P})$ among peers. It gives the mapping functions described for p_1 (the cost and address attributes) and for p_2 (the telephone number and country code). If there is a direct match between attributes, then the mapping function is empty ($\emptyset$).

B. *User-defined Views*

The idea behind user-defined views has been adapted from multiple data views in federated database research [21, 22] and these multiple data views have been proposed as an alternative for maintaining local autonomy within autonomous peers in P2P systems [13]. In our project, users are allowed to define data views, which comprise of peer preferences. These personalised views are based on the same shared schema and are manually created by the peer using the knowledge-base at the super-peer node. It provides information that will assist peers to forward their queries to the exact locations of required data sources.

V. CONCLUSION AND COMPARISON OF OUR APPROACH WITH CENTRALISED WEB SERVER SYSTEMS AND EXISTING SUPER-PEER SYSTEMS

Our proposed framework is significantly different from centralised web-server systems. The role of the super-peer is to help peers locate other assisting peers in order to facilitate the query forwarding coordination among peers, whereas the role of a server in a centralized web service is only to provide a directory of services provided by information providers.

Fig. 2. Part of an example knowledge base in RDFS format for the tourism ontology

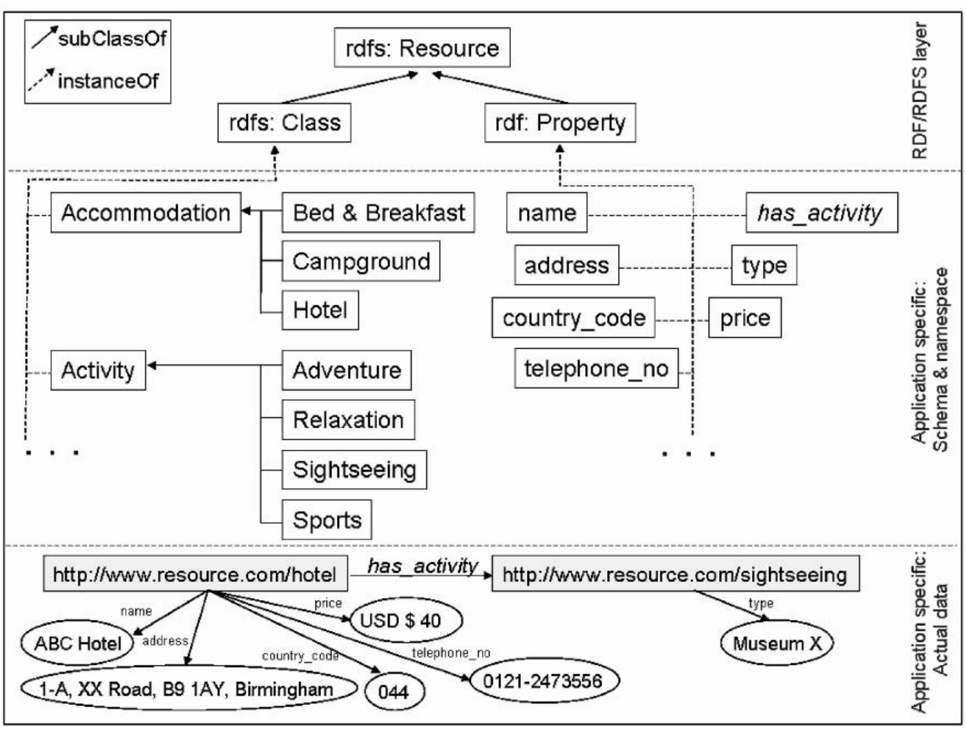


TABLE I: A SAMPLE OF SCHEMA RECONCILIATION

Ontology classes and attributes		**Peer p_1**	**Peer p_2**	…
Classes of $\mathcal{K}$ with subclasses indented	*Attributes of class/subclass*			
Accommodation	name, address, country_code, telephone_no, price			
Bed&Breakfast				
Campground				
Hotel		Hotel: name $\rightarrow p_1$: name, *mf*: $\varnothing$ Hotel: address $\rightarrow p_1$: address1 $\wedge$ p_1: address2 $\wedge$ p_1: postcode $\wedge$ p_1: city, *mf*: union Hotel: country_code $\rightarrow$ p_1: c_code, *mf*: $\varnothing$ Hotel: telephone_no $\rightarrow$ p_1: telephone#, *mf*: $\varnothing$ Hotel: price $\rightarrow p_1$: cost *mf*: poundToDollar	Hotel: name $\rightarrow$ p_2: name, *mf*: $\varnothing$ Hotel: address $\rightarrow$ p_2: address, *mf*: $\varnothing$ Hotel: country_code $\rightarrow$ p_2: telephone, *mf*: the first 3 digits Hotel: telephone_no $\rightarrow$ p_2: telephone *mf*: after the first 3 digits Hotel: price $\rightarrow p_2$: price *mf*: $\varnothing$	
• • •				

Our approach does not rely on the central server to carry out core tasks such as query answering. Instead, our super-peer node provides extra information that enables peers to perform ordinary P2P tasks even when the super-peer leaves the network. This contrast with the standard approach using a super-peer routing index, which is just a directory for the super-peer node to forward queries; query processing itself cannot be done without the super-peer node. Additionally, information providers are tightly bound to provide their information according to formats that have been setup by the server, whereas we just require a peer to map their local schema to the schema used in the shared knowledge in order to capture data source locations.

In centralised web server and existing super-peer systems, the query is processed using global data integration. Peers in our proposed framework are allowed to have a personalised user-defined view to assist them to get a query result with respect to their preferences. As a result, the proposed mediated knowledge-based view at the super-peer node is able to avoid unnecessary query results. In other words, the peers are able to frame queries so that they only return information of interest. Moreover, since the super-peer in our proposed framework does not perform query processing, it requires less computational power than the server in a centralised system and also existing super-peer network systems.

REFERENCES

[1] G. Koloniari and E. Pitoura. Peer-to-peer management of XML data: Issues and research challenges. ACM Sigmod, Vol. 34, No. 2, 34(2), June 2005.
[2] D. S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu. Peer-to-peer computing. Technical report, HP Laboratories Palo Alto, TR HPL-2002-57, 2002.
[3] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. What to ask to a peer: Ontology-based query reformulation. In Proceedings of the 9th Conference on Principles of Knowledge Representation and Reasoning (KR 2004), 2004.
[4] D. Tsoumakos and N. Roussopoulos. Analysis and comparison of p2p search methods. Proceedings of the 1st International Conference on Scalable Information Systems, 2006.
[5] Gnutella website. http://www.gnutella.com.
[6] M. Boyd, S. Kittivoravitkul, C. Lazanitis, P.J. McBrien and N. Rizopoulos. AutoMed: A BAV Data Integration

System for Heterogeneous Data Sources. In Proceedings of CAiSE04, Springer Verlag LNCS, 2004.
[7] A. Y. Halevy, Z. G. Ives, P. Mork and I. Tatarinov. Piazza: Data Management Infrastructure for Semantic Web Applications. Proceedings of the 12th International Conference on World Wide Web, 2003.
[8] W. Nedjl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmer and T. Risch. EDUTELLA: A P2P Networking Infrastructure Based on RDF, Proceedings of the 11th International Conference on World Wide Web, 2002.
[9] D. Calvanese, G. D. Giacomo, M. Lenzerini, R. Rosati and G. Vetere, Hyper: A Framework for Peer-to-Peer Data Integration on Grids, Proceedings of the international Federation for Information Processing (IFIP), 2004.
[10] Gnutella2 website. http://www.gnutella2.com/.
[11] A. Crespo and H. Garcia-Molina. Routing indices for peer-to-peer systems. In the Proceedings of the International Conference on Distributed Computing Systems (ICDCS), 2002.
[12] S. Gribble, A. Halevy, Z. Ives, M Rodrig and D. Siciu, What can databases do for peer-to-peer?, In Proceedings of the 4th International Workshop on the Web and Databases, 2001.
[13] N. E. Taylor and Z. G. Ives. Reconciling while tolerating disagreement in collaborating data sharing. Proceedings of the 2006 ACM SIGMOD international conference on Management of data, 2006.
[14] M. Arenas, L. Bertossi and J. Chomicki, "Consistent Query Answers in Inconsistent Databases", *In Symposium on Principles of Database Systems (PODS)*, 1999.
[15] D. Calvanese, G. D. Giacomo, M. Lenzerini and R. Rosati, "Logical Foundations of Peer-To-Peer Data Integration," *Proceedings of the 23rd ACM SIGMOD-SIGACT-SIGART*, 2004.
[16] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini and R. Rosati, "What to Ask to a Peer: Ontology-based Query Reformulation," *In Proceedings of the 9th Conference on Principles of Knowledge Representation and Reasoning*, 2004.
[17] G. Greco and F. Scarcello, "On the Complexity of Computing Peer Aggrements for Consistent Query Answering in Peer-to-Peer Data Integration Systems," *Proceedings of the 14th ACM International conference on Information and Knowledge Management (CIKM),* 2005.
[18] M. Uschold and M. Gruninger, "Ontologies: Principles methods and applications," *Knowledge Engineering Review*, pp.93–155, 1996.
[19] L. Mota, L. Botelho, H. Mendes, and A. Lopes. "O_3F: an Object Oriented Ontology Framework," *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, July 2003.
[20] F. M. Al-Wasil, W. A Gray, and N. J Fiddian. "Establishing an xml meta-data knowledge base to assist integration of structured and semi-structured databases," *In Proc. of the 17th Australasian Database Conference*, pp. 38–49, 2006.
[21] M. B. Al-Mourad, W. A Gray, and N. J. Fiddian. "Semantically rich materialisation rules for integrating heterogeneous databases," *22nd British National Conference on Databases (BNCOD 22)*, July 2005.
[22] D. D. Karunaratna. "Exploration of Semantic Information in the Creation of Multiple views of a Federated Database System," *PhD thesis Department of Computer Science Cardiff University*, May 2000.
[23] W. Nejdl, W. Siberski and M. Sintek. "Design Issues and Challenges for RDF and Schema-Based Peer-to-Peer Systems," ACM SIGMOD Record, Vol(32), September 2003.

User Perception on Intelligent Split Menu for Web Browser Data Entry

Saipunidzam Mahamad
Department of Computer and Information Science
Universiti Teknologi PETRONAS
31750 Tronoh, Perak, MALAYSIA
saipunidzam_mahamad@petronas.com.my

Syarifah Bahiyah Rahayu Syed Mansoor
Department of Computer and Information Science
Universiti Teknologi PETRONAS
31750 Tronoh, Perak, MALAYSIA
rahayum@petronas.com.my

Rajaletchume A/P Annamalai
Department of Computer and Information Science
Universiti Teknologi PETRONAS
31750 Tronoh, Perak, MALAYSIA
rajalech@yahoo.com

***Abstract*—The notion of intelligent split menu for web browser is to demonstrate the experimental domain model that is associated with a user interface to improve data entry. A split menu becomes popular recently in computer application. Now, it is extended into the web browser for data entry purposes. The objective of this study is to investigate on user perception of integrating intelligent split menu on web browser data entry. Therefore, a prototype web browser model is used to embed the web browser model with a data model in order to gather feedback on the user perceptions. Results were mainly positive, as many are keen on the idea of having split menu and integrating it on the web browser. It shows how a web browser can be enhanced to be more useful to the users.**

Keywords- Intelligent split menu, web browser data entry

I. INTRODUCTION

The technology shift has been so vast that it is impossible for us to catch up with it. The revolution of the World Wide Web has changed information distribution among people. One should consider human computer interaction elements in all aspects of computer interaction including web site interface. Besides, human computer interaction plays a major role in determining the usability of a website or web page. In this study, usability is how well we can achieve our goals in completing tasks or looking for information in web page.

It is proven that Graphical User Interfaces (GUI) such as menu bars, tool bars, and icons make a web page usable. GUI objects have a major function in determining the functions of the web page. Menu with selection-support offers user's faster access to frequently used functions. Well-designed selection-support menus facilitate learning for new users by offering a staged, guided path to support the discovery of available functions and useful resources [1, 2, 3].

In this paper, we study split menu functionality as data entry and navigation. According to Sears and Shneiderman [4], split menu placed two or three most frequently selected items are at the top of the menu and then the remaining items in a bottom section. These most frequent items are placed at the top of the menu as a hot list items. In addition, he proposed arranging the most likely options for easy selection by the users. This list arrangement is known as hot list, where he believes it can enhance human computer interaction. Another type of list is Hotbox. Hotbox combines several GUI techniques, which are generally used independently: accelerator keys, modal dialogs, pop-up/pull down menus, radial menus, marking menus and menu bars. These techniques are fitted together to create a single, easy to learn yet fast to operate GUI widget. The reason is it can handle significantly more menu items than the traditional GUI menu bar. Sears and Shneiderman found users prefer to use split menu because it provides most frequently items on the top section.

In this paper, the research was to improve the current architecture of the web browser by implementing split menu and hot list into the web browser's address bar. The motivation is to improve the efficiency of data entry by exploiting the intelligent split menu.

II. BACKGROUND AND RELATED WORK

The most common place to find split menu is the Microsoft Office font selection such as Microsoft Word 2000 as shown in Figure 1; items for frequently used fonts are located above the separator, and all others are below the separator. Despite the trivial algorithm for inferring the top section, the Word menu has some excellent characteristics where the anticipation of user input is unobtrusive; user control is maintained and has a high hit rate frequency with which one of the hot list elements is selected.

According to Dong Sock Lee [5], he claims that these menus make it easier to select high frequency items by giving them priority over low frequency items. It encourages first time user to choose correctly. He evaluated how each menu item

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 439–442.

caused a change in frequency distribution that the menu reflected changed. He altered the menu 2/3 to reflect a new menu distribution. In this experiment, the split menu has shown as the fastest overall performance menu implementation compared to traditional menu.

Figure 1. Split menu in Microsoft Word

Paula [7] showed between split menu, alphabetical menu and frequency menu, most users prefer split menu. Her first experiment of 73 individuals completing 12 tasks of choosing names was successful and showed adaptive interface is very important. It causes less error and completion time was faster which is better than static menu. In another experiment, she provided 100 tasks, the results shown split menu in terms of application were still the fastest but the same for error rate as her first experiment. Besides, Jim Warren [6] proved split menu to be a working product where top 20 lists for mouse click entry of ophthalmology diagnoses was the most used data entry method for doctors. The results were positive where the hit list is picked out all the time.

The users of the system found it to be more effective than the normal way of putting information [6]. For example, they have hot lists for RFE (reasons for encounter – symptoms, complaints, checkups and prescription). They believed the hot listing would minimize the typing done and maximize the ability to specify items from long lists with point and click methods. Moreover, it should be used in many applications.

In another research done by Mona Tom [8], she claims that split menu can be used in automobile multimedia application. Menu designs here is challenging because it has to be optimized so the driver can easily accomplish the desired task while controlling the vehicle. Menu interfaces for automotive environment must promote a rapid search and selection process where the user intuitively knows where to find and activate a specific menu item within the menu structure. During the horizontal motion, it is important that the cursor movement does not move away too much in the vertical direction and leave the parent item, which will close the sub menu [1, 9].

III. METHOD

The prototype of intelligent split menu is shown in Figure 2. In order to embed the split menu into browser, first it has to be adapted according to user's needs and goals. User models are used to represent information collected about the user whose needs and goals are to be determined. A user model is an explicit representation of the properties of an individual user; it is used to reason about the needs, preferences or future behavior of the user.

The information about the model is collected by indirectly infer a user's preferences or goals through their interaction with the web browser model. The design developed appropriate method to allow user views and modifies the information in the user model. Although this may add additional complexity to a system that is attempting to simplify tasks through adaptive techniques, there are substantial benefits where users will be better able to understand the system has automated actions and correct errors in the model that cause infuriating performance.

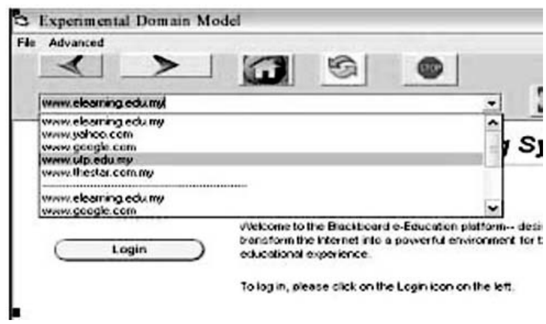


Figure 2. Split menu in Web Browser Model

A. Web Browser Model

A web browser model is a model which integrating an intelligent split menu on the web browser architecture. The split menu is applied on the web browsers address bar together with some basic functions such as stop, go, refresh, save and print.

The address bar is using dynamic drop down list. The top menu is the most frequently used address and the bottom ones are arranged in alphabetical order. To capture the data, a database, Microsoft Access, stores the address as entered by users. A counter is used to calculate the number of times user visiting a particular web site. In this model, the top five addresses will be listed as the hot list where it automatically arranges in the web browser inside the split menu.

B. Evaluation

The experiment was conducted at Universiti Teknologi PETRONAS (UTP), Malaysia. There were 50 students and staff involved in the end-to-end testing to evaluate the web browser model. The participants were then interviewed informally. After the interview session, they were required to fill up the evaluation questionnaires to justify their perception toward the web browser model. At the same time all the respondents were given the opportunity to browse the Internet using the web browser model. No specific time was given for each respondent. The questionnaires were designed in 3 main sections. Section 1 focused on users' satisfaction towards the system's functionality, which was the intelligent split menu. In section 2, provided the opportunity for the users to respond towards the web browser model. Section 3 focused on user responds toward implementation of web browser model. The user's perception on web browser model is evaluated from the questionnaire and not their behavior while using it.

IV. RESULT AND DISCUSSION

The response from various individual is different, as some did not understand the split menu terminology. Irrespective of their initial views, most respond with questions about the nature, format and the purpose of split menu in a web browser.

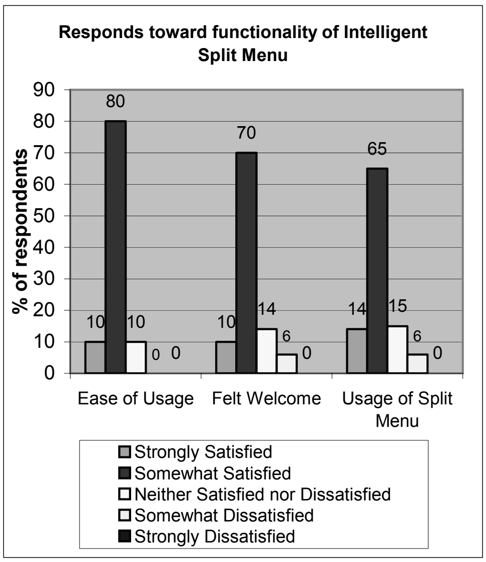


Figure 3. Responds toward functionality of Intelligent Split Menu

Figure 3 shows the result of the data analysis from 50 respondents towards section 1, the functionality of intelligent split menu. It showed a coherent finding that 80% of the respondents were satisfied with the ease of usage, felt welcome and usage of split menu towards functionality of intelligent split menu. The respondents felt that accessing Internet and find something much more easily through the web browser model. Regardless of that, 20% of the respondents somewhat were dissatisfied with the usage of split menu and rather to used current web browser

Figure 4 associates the result of section 2, responses towards the perception of Web Browser Model. Overall, the majority of the respondents chose 'Somewhat Agree' with all the statements. Statement on "Web Browser Model suits preference, enhance surfing experience and enhance HCI" showed the highest number, 60% and above of the total respondents. The respondents' felt that the functionality of split menu has improved their experience on accessing Internet. Meanwhile statement on "Web Browser Model increase interest" showed that 50% of the respondents 'Somewhat Agree' with this statement. Besides, 50% of the respondents 'Strongly Agree' the respondents believe using split menu and hot list may reduce typing and scrolling time for the address. Even though 10% of respondent 'Somewhat Disagree' with statement on "Web Browser Model enhances surfing experience" but it can be concluded that most of the respondents agreed with all the statements.

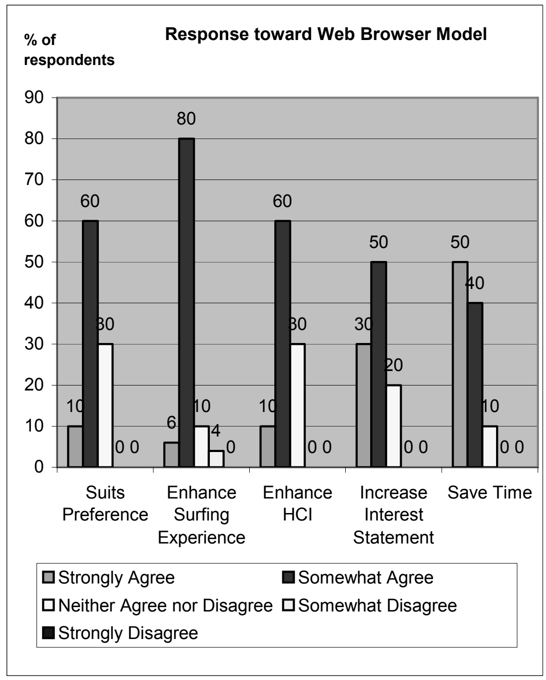


Figure 4. Responds to Web Browser Model

The result of section 3 is shows in Figure 5. There are 30% of the respondents were, 'Strongly Agree' and 50% of the respondents were 'Somewhat Agree' with implementation of Web Browser Model. However, 20% of respondents were "Neither Agree nor Disagree" are rigid towards changes and rather to use the current browser.

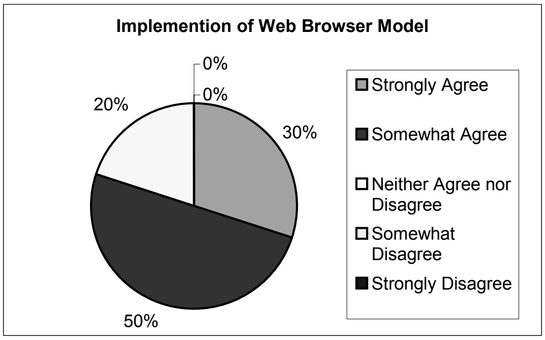


Figure 5. Implementation of Web Browser Model

V. Conclusion

Information is wealth like claimed by many people and time is gold. As entering the new paradigm, the search for information on various subject matters becomes important. This web browser model with the split menu feature is hoped to satisfy the user's needs and goals and at the same time satisfy the needs for fast information on the World Wide Web. It is appropriate that the controlled testing that has been done was provides a valuable support for the split menu. After this testing, users might expose to the benefits of the split menu in storing information.

It is advisable for interface designer to implement split menu while designing adaptive interfaces. From the testing and evaluation done, it can be concluded that by using intelligent split menu, user were satisfied in surfing through web browser model thus enabling them to adapt knowledge and awareness of user interface technology. It shows time can be reduced for the people who have to do repetitive task of choosing menus as they can choose the menu faster. Overall, the study has shown significant result and can be conclude that users are satisfied with the idea of implementing intelligent split menu for a web browser address bar as a data entry and had achieved its objective to improve the web browser architecture.

References

[1] Kath Straub, Adaptive Menu Design, *US Design Newsletter Insights From Human Factors International*, 2004.

[2] James R. Warren & Patrick Bolton, Intelligent Split Menus for Data Entry: A Stimulation Study in General Practice Medicine, *Inform Assoc, Symposium Supplement,* November 1999, pp. 450-454

[3] David Ahlstrom, Modeling and Improving Selection in Cascading Pull-Down Menus Using Fitts' Law, the Steering Law and Force Fields, *Proceedings of the SIGCHI conference on Human factors in computing systems Publisher: ACM Press*, 2005.

[4] A. Sears and B. Shneiderman, Split menus: Effectively using selection frequency to organize menus, *ACM Transactions on Computer-Human Interaction*, Vol. 1, No. 1, pp. 27-51, 1994.

[5] Lee, Dong-Seok, and Yoon, Wan Chul, Quantitative results assessing design issues of selection support menus. *International Journal of Industrial Ergonomics*, 33, pp. 41-52. 2004.

[6] Warren, J. Cost/benefit based adaptive dialog: case study using empirical medical practice norms and intelligent split menus; *Proceedings. Second Australasian*, 2001, pp 100 - 107

[7] Paula Selvidge, Menu Design: To Adapt, or Not to Adapt, http://psychology.wichita.edu/surl/usabilitynews/41/adapt_menus.htm 2002

[8] Mona L. Toms, Mark A. Cummings-Hill and David G. Curry**,** Delphi Delco Electronics Systems, Using Cluster Analysis for Deriving Menu Structures for Automotive Mobile Multimedia Applications, 2001

[9] Marc H. Brown, Robert A. Shillner, University of Princeton, DeckScape: An Experimental Web Browser (1995)

From Information Wiki to Knowledge Wiki via Semantic Web technologies

Sergiu Dumitriu, Marta Gîrdea, Sabin C. Buraga
Faculty of Computer Science, "A.I.Cuza" University of Iaşi
16 Berthelot, 700483 Iaşi, Romania – Email: {sergiu.dumitriu, marta, busaco}@infoiasi.ro

Abstract — **The paper presents an enhanced version of an existing wiki platform (XWiki), in order to integrate knowledge, based on various Semantic Web technologies. In particular, we describe how metadata, microformats, ontologies are meaningfully used in this context, and we show the utility of our approach via two use cases.**

I. Introduction

The World Wide Web space is primarily compounded by pages (documents that contain markup) with information in the form of natural language text and multimedia, intended for humans to read and to understand. Computers are principally used to render this hypermedia information, not to reason about it. However in the next stages of Web evolution, information is no longer intended only for human readers, but also for machine processing, enabling intelligent information services, personalized Web sites, and semantically empowered search engines – this is the seminal idea of the *Semantic Web* [4, 7]. The Semantic Web is viewed as "an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [4].

When advancing towards Semantic Web, the main obstacle is the effort that the creator of hypermedia information must put into organizing the knowledge and metadata, into tagging entities and relations, using vocabularies he must be familiar with, in order to make it comprehensible not only for humans, but also for machines. A possible answer to this problem could be a platform which helps *transparently* organize data and metadata for machine-comprehensibility, so that the contributors do not need to know Semantic Web vocabularies, as the system automatically generates metadata based on the creator's actions and on the progress of the information manipulated within the platform.

In order to create the knowledge base on a certain domain, one can use the power of online communities. The predominant type of collaborative Web application is represented by wikis [10]. For example, major successes of the wiki concept are the well-known Wikipedia [29] and its related initiatives.

The established wiki systems provide a basic Web interface for (collaborative) content editing by using a simplified syntax. Main facilities are unrestricted editing, the rollback mechanism, several search functions, the support for uploading content. Wiki platforms are currently used for different purposes, such as encyclopedias, collaborative writing, personal or public information management, project management, and many others [14].

To achieve significant knowledge acquisition, as opposed to simple data acquisition, a modern wiki system must provide a means of adding metadata about the concepts and relations established between the concepts within a given wiki.

The goal of this paper is to detail a semantic Web-based extension of the XWiki platform, which can be used to present, create and manage knowledge by both experienced and inexperienced users.

After a short presentation of the current initiatives towards Semantic Wikis, in section III we detail the general architecture of the XWiki system. Section IV is dedicated to the semantic Web-based enhancements of the XWiki platform, followed by two case studies. In the final section, the paper concludes and anticipates further directions of research and development.

II. Context, Related Work and Initiatives

A. Short Presentation of Semantic Web Technologies

Semantic Web technologies are based on the XML [5] family of markup languages and are structured on three main layers [7]:

- *Metadata* layer offers an extensible framework in order to express simple semantic assertions. This conceptual model can be use to attach metadata to resources. Using metadata, we can formulate statements about certain Web resources in the form of <*subject*, *predicate*, *object*> triples. There are available several metadata vocabularies such as DCMI (Dublin Core Metadata Initiative) [17], FOAF (Friend of a Friend) [27], or XFiles [6].
- *Ontology* layer defines a hierarchical description of the concepts and properties of a given resource, from

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 443–448.

simple taxonomies to complex upper-level ontologies.

- *Inference* layer offers the possibility to reason about knowledge by using rules.

Several XML-based languages specialized in the modeling of knowledge were developed in the context of Semantic Web: RDF (Resource Description Framework) [11] and OWL (Web Ontology Language) [8]. Ontological classes are denoted by OWL assertions, and their instances (individuals) are expressed by RDF statements.

A different direction is the use of microformats [22], with the goal of embedding semantic data directly within documents conforming to a presentational XML format (especially XHTML). For example, we can embed complex taxonomies and schemas on the basis of the markup of regular Web pages (e.g., blog entries, wiki documents, etc.). With the help of microformats and appropriate vocabularies we can denote personal virtual cards, tags, relations between persons, events and calendars, etc. This approach is currently used in the creation of the Social Web, one of the facets of "Web 2.0" [13].

B. Semantic wikis

Several initiatives of wiki platforms which provide support for expressing knowledge via current Semantic Web technologies are already available. Most of them are simply ontological editors in the wiki style, which can not be used by people without a prior knowledge of the (Semantic) Web.

The *Rhizome* [15] system offers support for capturing and representing informal, human-authored content and is based on Semantic Web languages. Rhizome Wiki uses custom text formats to make writing semantic content easier, which can be managed by a RDF data access engine. Unfortunately, the rendered content does not contain embedded rich metadata or microformats and final users do not fully benefit of RDF-like stored knowledge. Also, Rhizome's support for expressing ontological constructs is very basic, and it requires the users to be familiar with XML and related technologies.

One of the promising initiatives is *Semantic MediaWiki* [9, 24], a prototype built on the MediaWiki platform, using an external RDF repository. A possible use of such a system is to facilitate knowledge acquisition [12].

Another Semantic Wiki for collaborative knowledge management is *IkeWiki* [14], which supports different levels of formalization (from informal texts to formal ontologies expressed in OWL) and a rich user interaction – e.g., SVG (Scalable Vector Graphics) and AJAX (Asynchronous JavaScript And XML). IkeWiki is developed in Java and uses the well-known Jena [21] semantic platform. IkeWiki presents to users links to (semantically) related pages and provides a tool for automatic generation of content from existing semantic annotations (DCMI metadata or RDF statements).

In order to ease the ontological content creation in a collaborative manner, an interesting platform is *OntoWiki*, based on pOWL [2]. According to [3], *OntoWiki* is a tool that integrates RDF triples into the wiki textual content by using a special syntax in order to simplify the presentation and acquisition of instance data from and for end users in a generic way.

We noticed the majority of available semantic wiki systems can be used by specialists only and can not address the needs of regular communities to exploit the power of semantic Web technologies in a pragmatic way.

III. Architecture of the XWiki Platform

A. Description

We based our implementation on the XWiki platform [28], a powerful open source wiki engine written using J2EE technologies. XWiki is a professional wiki, having many features needed for enterprise usage. It is also an application wiki by allowing development of applications with structured data and scripting right from the Wiki interface. It can be used in a large variety of scenarios, from a simple presentation site, a personal wiki or blog, to a complex structured content repository or advanced collaborative application.

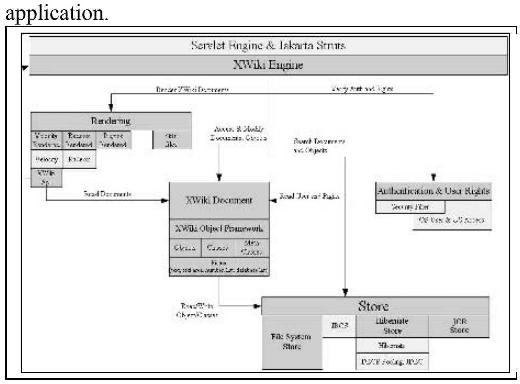


Fig. 1. Overall XWiki architecture.

XWiki is inspired by TWiki [25], a wiki engine written in Perl. The initiator and maintainer of XWiki is Ludovic Dubost, leading a growing team of developers. The first version of XWiki was made public in January 2003, and the official 1.0 version is planned for early 2007.

B. General Structure

The power of XWiki lies in the multi-layered architecture, separating the platform into distinct, loosely coupled modules, based on the MVC (Model-View-Controller) design pattern. The most important modules are depicted in Figure 1, and are described below:

- XWiki is deployed as a *servlet*-based web application, taking full advantage of the underlying server API, handling multithreading, HTTP requests and responses, thread pooling, etc.
- The *XWiki engine* ties all the other modules together, serving as an *intermediary*. It offers APIs that can be

used from both the Java code and the web rendering modules, described below.

- Documents are the main information entities, and are managed by the *XWikiDocument* module. A document can be used as a simple page container, but it is possible to use such documents to define structured information units or mini-web applications, through the use of a Class API, attached Objects, presentation templates and velocity or groovy in-line scripts. This is the *Model* part of the well-known MVC pattern.
- The *rendering module* consists of multiple rendering engines executed in a cascading process. The main rendering engines are Velocity [26] and Groovy [19], which execute scripting code included in the document content. Also, XWiki uses the Radeox [23] renderer, a lightweight wiki rendering engine, and a set of configurable plug-ins which perform various processing tasks. The rendering module, together with the skin files (e.g., CSS), form the *View* part in MVC.
- The *store module* is responsible for persistent storage and document versioning. Either Hibernate, as an object to relational DBMS intermediary, or JackRabbit as a full-featured Java Content Repository can be used. For performance reasons, a Cache module handles in-memory and on-disk caching.
- The *Authentication and Rights Management* module is responsible for the user and group rights (which concern viewing, editing and programming rights) over documents.

C. Data Organization

The information is organized into Documents, Objects, Classes and Attachments.

The Document is the entity that specifies the displayed content. It can contain raw text, wiki or (X)HTML markup, and interpreted scripts. Each document is described by properties (metadata), such as the document creator or modification date.

A document can be used to host a *Class* by attaching class properties. Classes model the structure of some target information stored in the wiki. The power of these entities is limited by the fact that no relations can be defined between classes (such as inheritance).

Objects are instantiations of classes, attached to the document they were created in. Some document objects define structured metadata related to the document (for example, comments and page access rights). Objects can also be used to define the actual document content as a structured information unit, such as the entities defining users, groups, preferences or blog entries.

Attachments are files uploaded by users and whose content is stored in the database. The platform defines metadata for each attachment, such as upload time, file size or creator.

D. User Interface and Presentation

Editing information units in XWiki is aided by the simple wiki syntax, a friendly WYSIWYG editor, and various other AJAX-based tools. Structured documents can be built using a class definition, a template from which the document contents and attached objects are copied, and a sheet which transforms the document editing into a form completion task (in-place editing).

A number of velocity template files control how the document is rendered. The velocity code can use the XWiki API to manipulate the documents and their attached metadata (properties, objects, attachments), and can use the API that each plug-in defines. The content of a document is formatted by such templates for Web rendering, portlet integration, PDF or RSS (Really Simple Syndication)/Atom feed generation, etc.

IV. From Information Wiki to Knowledge Wiki

A. Context

XWiki manages internally a large range of well-structured information and metadata whose purpose regards mostly the user's benefit. However, this rich information is not fully valued. It is displayed as raw data, along with all the other contents of the documents, and cannot be automatically identified and used by a machine, despite the well-defined structure.

When advancing towards Semantic Web standards [30] compliance, such information should be presented in machine-comprehensible vocabularies. This section describes the metamorphosis of the user friendly information wiki into a user and machine friendly knowledge wiki.

B. Support for Metadata

1) DCMI (Dublin Core Metadata Initiative)

For each document, the creator and the creation and modification date are usually displayed using plain HTML markup in a dedicated area of the page. To make this information semantically accessible, this information, along with other document properties, can be displayed using the DCMI vocabulary [17] using meta-tags in the document header. The same information can be displayed using an external RDF document describing the document. An example of DCMI constructs to express the document attributes follows:

```
<link rel="meta" href="/bin/rdf/Programs/DoctoralSchool" />
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<meta name="DC.title" content="Doctoral School"/>
<meta name="DC.creator" content="Adrian Iftene"/>
<meta name="DC.contributor" content="Sabin Buraga"/>
<meta name="DC.created" content="2006-09-11"/>
<meta name="DC.modified" content="2006-10-04"/>
<meta name="DC.format" content="text/html"/>
<meta name="DC.language" content="en"/>
```

2) Microformats

Some of the predefined classes of XWiki, namely the ones describing Users, News and Events, present well structured information units, for which microformats can be attached. Since microformats require only separating the different parts of the description and specifying the *class* attribute, adapting XWiki to display *hCard* and *hCalendar* [22] requires minor modifications to the sheets that render user pages and event objects, with no impact on the visual layout of the information. The below example and Figure 2 illustrate the use of *Tails* extension for Firefox, which can automatically detect and extract *hCard* microformat information about a person (in this case, a PhD student).

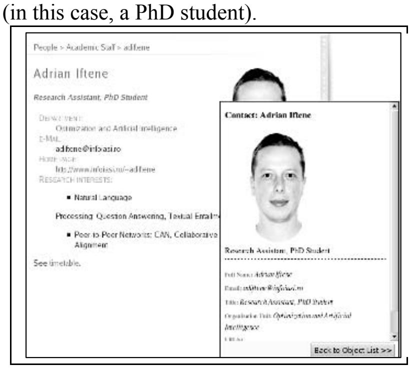


Fig. 2. *Tails* recognizes the hCard included in the page.

```
<div class="vcard">
<img src="/fii/bin/download/XWiki/adiftene/adrian_iftene.jpg"
   alt="XWiki.adiftene" width="192" class="photo"/>
<h2 class="heading-1"><span class="fn">Adrian Iftene</span></h2>
<h4 class="heading-1-1-1">
 <span class="title">Research Assistant, PhD Student</span></h4>
<dl>
 <dt>Department:</dt>
 <dd><a href="/fii/bin/view/Structure/OIA" class="organization-
unit">Optimization and Artificial Intelligence</a></dd>
 <dt>e-Mail:</dt>
 <dd class="email">adiftene@infoiasi.ro</dd>
...
</dl>
</div>
```

Figure 3 denotes the rendering of event information expressed by *hCalendar* microformat constructs.

In the same manner, other microformats can also be implemented, like *adr*, *hResume*, or *xFolk*. Moreover, each presentation sheet can be organized in a semantic way, thus defining a custom microformat.

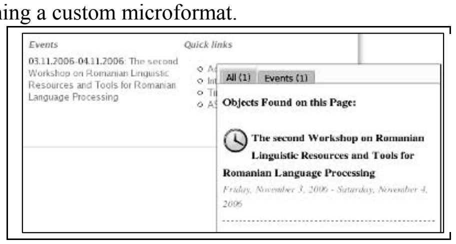


Fig. 3. Event information embedded into the XHTML code of a XWiki page.

3) Using FOAF

The user profile is displayed using plain markup, making it difficult for an agent to detect the fact that a page actually describes a person. For such information, the FOAF (Friend of a Friend) [27] initiative created a set of standard properties that can be used to describe a person, his/her activities and the relations with other people. FOAF properties can be used inside RDF to describe an entity. While it is hard to automatically deduce the relations among people, simple properties can be generated from the object defining the user, using a custom presentation template.

An example of FOAF construct follows:

```
<rdf:RDF
 xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
 xmlns:foaf="http://xmlns.com/foaf/0.1/">
<foaf:PersonalProfileDocument rdf:about="/bin/view/XWiki/adiftene">
 <foaf:maker rdf:resource="#adiftene"/>
 <foaf:primaryTopic rdf:resource="#adiftene"/>
</foaf:PersonalProfileDocument>
<foaf:Person rdf:ID="adiftene">
 <foaf:name>Adrian Iftene</foaf:name>
 <foaf:givenname>Adrian</foaf:givenname>
 <foaf:family_name>Iftene</foaf:family_name>
 <foaf:homepage rdf:resource="http://www.infoiasi.ro/~adiftene/"/>
 <foaf:phone rdf:resource="tel:+40232201553"/>
</foaf:Person>
</rdf:RDF>
```

4) Semantic Links

Most semantic wiki systems provide support for link descriptions. For this purpose, we extended the Radeox [23] filter, which transforms the wiki syntax for links into (X)HTML anchors, to accept two more parameters, rendered as the content of the *title* and *rel* attributes, respectively. Such a feature obviously allows defining links "with a meaning" (as an opposite to plain linking, which is almost exclusively used in hypertext). An advanced web crawler can gather information about the linked resources (for example, in a page about a "book" resource, one can detect which of the links point to book chapters, which points to the author, the publisher, etc.). A special use of the *rel* attribute defines the *rel-tag* microformat [22].

5) Support for Folksonomies

We defined a custom class to represent *tags*, which bind keywords to a page. Each user can attach tags to resources, allowing users to find similar pages, browse pages according to tags, and obtain better results for keyword search in a social Web tools manner [13] (e.g., del.icio.us, Flickr.com, etc.).

C. Modeling Knowledge via RDF and OWL

The powerful class-based infrastructure of XWiki, which allows structuring information within the wiki, lead to the idea of further developing the semantic organization of information. The first step we made is the extension of the class definition in order to accept inheritance, by adding the *base* property, where a multiple selection list allows choosing the base classes from all the classes existing in the system.

Since XWiki can organize information into objects as instances of a class, it is natural to view each object as an entity, described as RDF triples. For each class, an OWL document can be generated, in order to capture knowledge regarding the wiki content. Each object (instance of a certain class) has a RDF file binding the object definition to the given OWL class. More details about the ontological constructs are explained by examples in the next section.

Our approach offers support for knowledge representation through the modeling already existing in the XWiki platform. We give an ontological meaning to this modeling, previously used only internally.

V. Case Studies

We deployed the semantically enhanced wiki platform in the context of e-learning: a repository of didactic materials and the Web site of the Computer Science department of our university, which are briefly described below.

A. Managing E-learning Resources in the Wiki Style

The aim of the ITeach (Innovative Teacher) [20] project is to develop practical methodologies, approaches and tools targeted to day-to-day utilization of enhanced ICT skills in the work of trainers and teachers. The ITeach wiki repository provides an easy to use interface for creating scenarios and tasks concerning the usage of different ICT technologies in the teaching process.

The specification for the repository requested the possibility of maintaining relations between the stored objects. Thus, an object can be translated, copied or adapted into another. We expressed these relations using the DCMI vocabulary for metadata attached to each didactic material resource available within the platform:

```
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<meta name="DC.title" content="Sample Task"/>
<meta name="DC.description" content="This is a sample task, created during the training program."/>
<meta name="DC.creator" content="Mihaela Brut"/>
<meta name="DC.contributor" content="Italo Calvino"/>
<meta name="DC.contributor" content="Vladimir Ivanov"/>
<meta name="DC.created" content="2006-09-26"/>
<meta name="DC.modified" content="2006-09-27"/>
<meta name="DC.format" content="text/html"/>
<meta name="DC.language" content="en"/>
<meta name="DC.isPartOf" content="Sample Scenario"/>
```

Also, the *hCard* microformat was used, as mentioned above, for creating virtual cards for the teachers that collaborate for the development of the repository.

Metadata is the key component of a semantic empowered search engine available within the platform.

B. Academic Organization Model

This case study concerns the Web site [18] of our department, built on the XWiki platform.

The entities that participate to the academic life in a computer science department are persons (professors, students, and auxiliary personnel), specializations (undergraduates, graduates, and postgraduates of each age group), course curricula, and other types of resources (e.g., geographic locations, time). These concepts form a portal ontology, similar to AKT ontology [16], but more adequate for our purposes.

Each wiki entity is modeled as a class (for example, *Professor* derived from *User*, *Department*, *Generation*, *Course*, *OptionalCoursePackage*, etc.). Every class instance has different associated metadata exported as RDF statements. Several properties establish relations between different kind of objects (e.g., a "Professor" *teaches* a "Course" and a "Course" *isTaughtTo* a "Generation" of students). This approach helps us to offer a proper navigational support within the wiki and minimizes redundant data. For example, in Figure 4., the user has access to a list of quick and related links, automatically generated according to document relations.

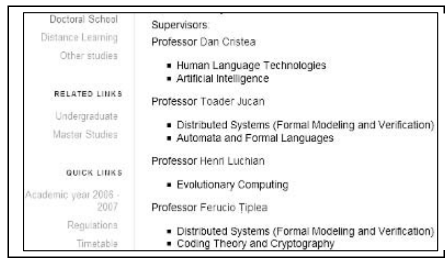


Fig. 4. Navigational Structures provided conforming to document relations.

Our platform offers an easy to use WYSIWYG editor of content, including different associated semantic data (metadata, properties, relations, etc.). As an example, Figure 5 presents the editing form for an object belonging to the *Course* class. This activity can be accomplished by authenticated users via the interface of the Web site.

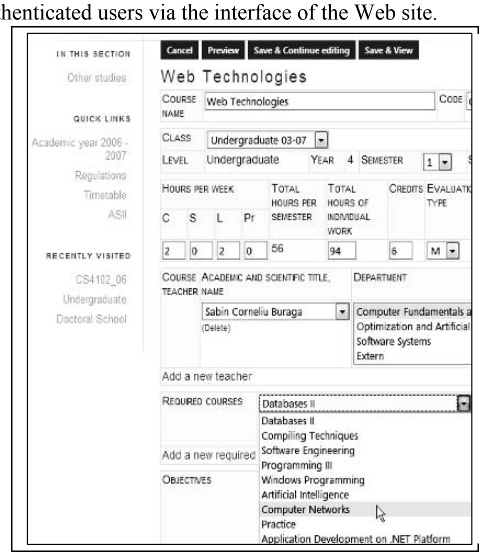


Fig. 5. Inline editing of a course description (metadata is embedded).

In contrast with the closed-world approach (where all classes and properties are *a-priori* defined), XWiki permits adding and modifying classes and the respective properties at any moment.

All changes can be traced by subscribing to the provided RSS/Atom feed. In Figure 2., we already presented the use of microformats within the site.

VI. Conclusions and Further Work

The paper presented an extension to XWiki that adds support for knowledge representation and manipulation, aligned to current practices of the Semantic Web. XWiki was enhanced with microformats, tagging, semantic links, and metadata export to simple formats like DCMI and FOAF. Furthermore, the entire document class infrastructure of XWiki can now be exported to standard RDF and OWL. Two case studies illustrate the viability of our solution in the context of online collaboration within organizations.

With the wider adoption of XHTML 2.0 [30] in the near future, XWiki will be able to generate output using this language, so that the semantic information will be more easily exposed and used directly into the rendered markup code.

Another direction of interest is adding the ability to import RDF and OWL as XWiki objects and classes respectively.

Further directions of research are focused on integrating the Jena engine [21] into the XWiki system (as an external plug-in or module), in order to offer reasoning services in a transparent manner. At the moment, a kind of rule-driven behavior is provided by scripts and internal SQL queries.

References

[1] B. Adida, M. Birbeck (eds.), *RDFa Primer 1.0. Embedding RDF in XHTML*, W3C Working Draft, 2006: http://www.w3.org/TR/xhtml-rdfa-primer/

[2] S. Auer, “pOWL: A Web-based Platform for Collaborative Semantic Web Development”, *Proceedings of the Workshop on Scripting for the Semantic Web*, Heraklion, Greece, 2005.

[3] S. Auer, S., Dietzold, T., Riechert, “OntoWiki - A Tool for Social, Semantic Collaboration”, *Proceedings of 5th International Semantic Web Conference*, Lecture Notes in Computer Science – LNCS 4273, Springer-Verlag, 2006, pp. 736–749.

[4] T. Berners-Lee, J. Hendler, O. Lassila, “The Semantic Web”, *Scientific American*, 5, 2001.

[5] T. Bray et al. (eds.), *Extensible Markup Language 1.0 (Fourth Edition)*, W3C Recommendation, Boston, 2006: http://www.w3.org/TR/REC-xml

[6] S. Buraga, “A Model for Accessing Resources of the Distributed File Systems”, *Advanced Environments, Tools and Applications for Cluster Computing*, Lecture Notes in Computer Science – LNCS 2326, Springer-Verlag, 2002, pp. 224–230.

[7] M. C. Daconta, L. J. Obrst, K. T. Smith, *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*, John Wiley & Sons, 2003.

[8] M. Dean, G. Schereiber (eds.), *OWL Web Ontology Language Reference*, W3C Recommendation, Boston, 2004: http://www.w3.org/TR/owl-ref/

[9] M. Kroetzsch, D. Vrandecic, M. Voelkel, “Wikipedia and the Semantic Web – The Missing Links”, *Proceedings of Wikimania 2005*, Frankfurt, 2005.

[10] B. Leuf, W. Cunningham, *The Wiki Way. Quick Collaboration on the Web*, Addison-Wesley, 2001.

[11] F. Manola, E. Miller (eds.), *RDF (Resource Description Framework) Primer*, W3C Recommendation, Boston, 2004: http://www.w3.org/TR/rdf-primer/

[12] I. Millard, A. Jaffri, H. Glaser, B. Rodriguez, “Using a Semantic MediaWiki to Interact with a Knowledge-based Infrastructure”, *The 15th International Conference on Knowledge Engineering and Knowledge Management – EKAW 2006*, Podebrady, Czech Republic, 2006.

[13] T. O’Reilly, *What Is Web 2.0*: http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[14] S. Schaffert, A. Gruber, R. Westenthaler, “A Semantic Wiki for Collaborative Knowledge Formation”, *Proceedings of Semantics 2005*, Vienna, Austria, 2005.

[15] A. Souris, “Building a Semantic Wiki”, *IEEE Intelligent Systems*, vol. 20, no. 5, pp. 87–91.

[16] * * *, *AKTors.org*: http://www.aktors.org/akt/

[17] * * *, *DCMI (Dublin Core Metadata Initiative)*: http://www.dublincore.org/

[18] * * *, *Faculty of Computer Science Website*, “A.I.Cuza” University of Iasi, Romania: http://www.info.uaic.ro/

[19] * * *, *Groovy*: http://groovy.codehaus.org/

[20] * * *, ITeach Repository: http://i-teach.infoiasi.ro/

[21] * * *, *Jena*: http://jena.sf.net/

[22] * * *, *Microformats*: http://www.microformats.org/

[23] * * *, *Radeox*: http://radeox.org/

[24] * * *, *Semantic MediaWiki*: http://meta.wikimedia.org/wiki/Semantic_MediaWiki

[25] * * *, *TWiki*: http://twiki.org/

[26] * * * *Velocity*: http://jakarta.apache.org/velocity/

[27] * * *, *FOAF (Friend Of A Friend) Vocabulary Specification*: http://xmlns.com/foaf/0.1/

[28] * * *, *XWiki*: http://www.xwiki.org/

[29] * * *, *Wikipedia*: http://www.wikipedia.org/

[30] * * *, *World-Wide Web Consortium’s Technical Reports*: http://www.w3.org/TR/

Content Based Video Retrieval Framework Using Dual Tree Complex Wavelet Transform

Tahir Jameel, S.A.M. Gilani, Adeel Mumtaz
Faculty of Computer Science and Engineering
Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan
saroash_khan@msn.com, {teejay, asif, adeel}@giki.edu.pk

Abstract— **In this paper a novel technique of content based video retrieval is presented. The proposed technique uses Dual Tree Complex Wavelet Transform (DTCWT) based features of video frames for the purpose of shot change detection, key frame selection and video indexing. For shot change detection consecutive frame difference is computed, shot change is reported when the difference exceeds a certain threshold. For keyframe selection a frame is to be selected which is not part of shot transition using k-mean clustering of DTCWT feature vectors. Video shots are indexed using DTCWT features of the selected keyframes. Video query is processed by comparing the features of shot with the features database of the shots. For the purpose of features similarity we have used correlation based distance metric as it produced better results for this kind of feature similarity. The results are compared the results with classical techniques and it is shown how dual tree complex wavelet transform based features performed better. The whole framework uses similar kind of feature which makes it simple and efficient.**

Index Terms— **CBVR, Video Indexing, Shot Boundaries, Key Frames**

I. INTRODUCTION

A PICTURE is worth a thousand words is famous proverb. If it is so then video must worth a million words. Video has become a very useful source of multimedia information containing rich, self explanatory and most compelling information presentation source. In this era a revolution has taken place that has transformed the concepts of imaging to the digital media technology paradigm. Enhanced networking capabilities and digital technology has changed physical management of information to an interactive and more effective electronic data management. A large variety of digital video collection is available online about education, entertainment, business, current affairs and medicines. Different video formats are available for compression and video storage. Digital video is an aggregation of frames where each frame is a picture image. These frames are presented sequentially with certain frame rate. This gives an illusion of motion picture. Typical frame rates are 25 or 30 frames per second [1]. With this frame rate we can expect storage required for an hour of digital video. Main concerns about digital video are storage, displaying and searching. Remarkable work has been done in display quality and compression with certain trade off. Most of recent research is being done in efficient search systems for digital videos. Digital video in the perspective of digital images naturally makes possible to apply image processing techniques for compression, storage and searching.

Vide data is hierarchy in structure as shown in Fig. 1. Video sequence is composed of scenes. A scene is a story unit and it is a collection of consecutive shots that have semantic similarity in object, person, space and time [1]. It can be understood as a semantic situation having contents of various location camera shoots. Scene is more towards human understanding because it's close to human feelings and perception. A scene can be a collection of shots. A shot is defined as a continuous camera capture in which a camera can move as well as the objects can move. It is a basic unit of video like a paragraph in a text document. Contents of a shot are quite similar. Ideally there is negligible difference among the frame of a shot. In content based retrieval system, data is usually stored retrieved and demanded in shots. So a video sequence is decomposed into shots and stored in database. Shot boundary detection is usually a pre-processing technique for the video indexing system. In hierarchy a single shot is a collection of temporal correlated frames. Ideally there is unnoticeable difference among the consecutive frames. To reduce redundant information in the shots and to make retrieval system efficient a single frame which can represent the contents of a shot is selected. Such frames are known as keyframes. A shot may contain more than such frames which are known as candidate frames. The technique is to group candidate frames and select a non transient frame. In MPEG [2] video format there are I, P and B frames. I frame is a complete image containing all its data blocks which are referenced in P and B frames. This I frame may be selected as keyframe.

In section II we have shown how DTCWT features are extracted. In section III it is shown that how DTCWT features dissimilarity is utilized for shot boundary detection. Section V narrates video indexing and query shot retrieval. In section IV k-means clustering is applied on DTCWT features for the purpose of key frame selection. Section VI shows the results and finally section VII concludes the whole discussion.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 449–454.

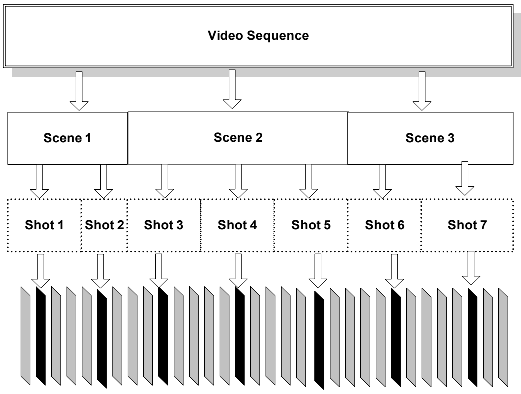


Fig. 1 Video Structure Hierarchy

II. CONTENT BASED VIDEO SIGNATURES

In most of the earlier techniques, video keyword based searching and text based searching have been prominent. But the obvious problem is that sometime keywords can not describe video contents, even the text based methods require tedious job to enter annotations manually. With advent of internet the problem of language have been raised in keyword based searches. It is possible that the most relevant video is available on the network but the language of text/ keyword is different. It was required to attach labels with the video which describe their visual contents. Trivial image description techniques have been applied to describe video contents. Most common are the color histograms, shapes, texture or motion based techniques. For video shot description histogram based techniques produced remarkable results because of temporal correlation, they are robust to object's geometric movements. Also frequency domain techniques have been proposed, amongst them wavelet based techniques have been significant [3] [4].

Kingsbury's [5] dual-tree complex wavelet transform (CWT) is an enhancement to the discrete wavelet transform (DWT), with important additional properties. The main advantages as compared to the DWT are that the complex wavelets are approximately shift invariant (meaning that our texture features are likely to be more robust to translations in the image) and that the complex wavelets have separate sub-bands for positive and negative orientations. Conventional separable real wavelets only have sub-bands for three different orientations at each level, and cannot distinguish between lines at 45º and -45º**.** The complex wavelet transform attains these properties by replacing the tree structure of the conventional wavelet transform with a dual tree. At each scale one tree produces the real part of the complex wavelet coefficients, while the other produces the imaginary parts. A complex-valued wavelet $\psi(t)$ can be obtained as:

$$\psi(t) = \psi_h(t) + j\psi_g(t) \tag{1}$$

where $\psi_h(t)$ and $\psi_g(t)$ are both real valued wavelets.

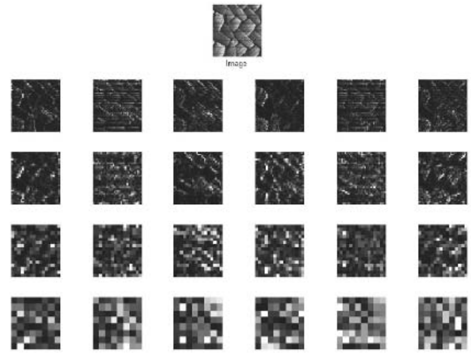


Fig.2. Four-scale CWT of a texture image

CWT like Gabor transform have six orientations at each of four scales (any number of scales can be used, but the number of orientations is built into the method). The main advantage as compared to the Gabor transform is speed of computation. It has a redundancy of only 4 in 2-dimensions and so the post-processing stages (of calculating mean and standard deviation) are also faster as it has less redundancy than the Gabor wavelets. Fig. 2 shows magnitudes of CWT coefficients for a texture image, one can see more details about orientation and scales. Each row represents one scale and columns represent angles within that scale. We performed a four scale (six angles) CWT on each frame. We get 24 real and 24 imaginary detailed sub-bands, and 2 real and 2 imaginary approximation sub-bands. By taking the magnitudes of corresponding real and imaginary coefficients of both approximation and detailed sub-bands we get 26 sub-bands. To calculate the features we measure the mean and standard deviation of the magnitude of the transform coefficients in each of 26 sub-bands, in the same way as [6].

III. SHOT BOUNDARY DETECTION SCHEME

Shot boundary detection is a pre-requisite of various content based video retrieval systems. A video shot is a video sequence that consists of continuous video frames for one camera action [7]. A video sequence can have more than one shot. These shots are merged together such that there boundaries are ill defined. Sometimes it is not obvious to detect the ending or start of a shot.

A number of video editing and mixing software are available to create special effects to merge shots which are known as shot transitions. Shot transitions are of various types [8]:

- A *cut* is an abrupt shot change that occurs in a single frame;
- A *fade-in* starts with a black frame; gradually the image of next shot appears, brightening to full strength.;
- A *fade-out* is opposite of a fade-in.;
- A *dissolve* consists of super imposition of a fade out over a fade in.

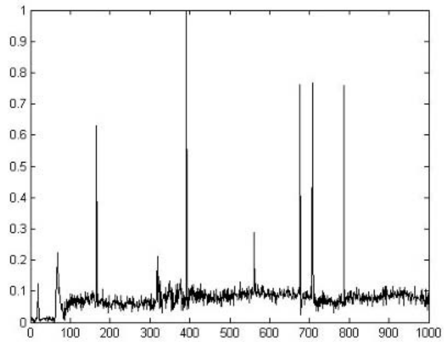


Fig. 3 Wavelet based Feature vectors Dissimilarity measure of first thousands frames of Challenge at Glen Canyon

Most of the shot boundary detection techniques are based on measuring consecutive frame differences. Different kinds of features have been used for shot change detection such as pixel level comparisons, histogram based techniques and DCT coefficients [8]. These techniques have some limitations regarding automatic threshold selection, light intensity change and corrupted frames like flashes. Flashes are due to different kind of noises during video capture. Dong Zhang [9] proposed a technique to counter these problems. A wavelet feature based technique was proposed by Satoshi Hasebe et al [4]. The feature vector used is based on coarsest subband and selected coefficients of higher energy bands. Fig. 3 shows consecutive frame's feature difference. The peaks in the graph identify the shot boundaries as there is significant change in the contents of the frame. Experiment is performed on a test sequence *Challenge at Glen Canyon*. It is a documentary of 27 minute including 48,450 frames. A threshold of 0.15 creates eight shots in the first thousand frames. But global threshold selection is a point to consider. We computed DTCWT features of each frame as explained in section II and computed consecutive frame distance using Euclidean distance metric. For test purpose we have used similar thousand frames of test sequence. There is one gradual and eight abrupt transitions in the experimental data.

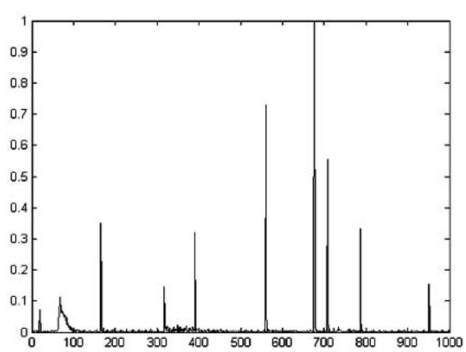


Fig.4 DTCWT features vectors dissimilarity of first thousand frames of Challenge at Glen Canyon

TABLE-1 RESULTS OF SHOT CHANGE FRAMES OF WAVELET BASED AND DTCWT

No. of Cuts	WAVELET	DTCWT
1	**66**	**68**
2	**165**	**165**
3	**318**	**317**
4	**391**	**391**
5	**561**	**561**
6	**677**	**677**
7	**708**	**708**
8	**787**	**787**
9	**NIL**	**951**

Fig. 4 shows the normalized difference of consecutive frames versus number of iterations. Table 1 shows a comparison of the proposed technique with previously discussed technique. It shows frame numbers where shot transitions are identified. It is shown that the proposed technique has produced nine transitions correctly. We need a threshold level to define the correct shot boundaries. It is seen that due to noise or flashes sometimes there are incorrect peaks. So threshold must be large enough to overlook these unnecessary peaks and should be sensitive enough to avoid shot detection misses. A threshold of 0.1 correctly produces 9 shots.

IV. KEYFRAME SELECTION

In content based video retrieval system keyframe selection is a vital task. A keyframe is an ambassador of a shot which describe its contents. Keyframes are also known as still abstracts of a video sequence. Apart from searching systems keyframe selection is a pre-requisite of digital video watermarking and storyboarding for short representation of digital video. Consecutive frames of a shot are quite similar to each other. To avoid redundancies a single key frame per shot is sufficient to summarize. Shot boundary detection techniques can not produce 100% results because of transition effects and flash lights. So the boundaries of the shot are not well defined. Due to mixing effects they have shot transition frames. While selecting keyframe from a single shot, it is obvious that frames near the boundary are likely to be part of shot transitions. The candidate key frames range lies between frames at boundaries. Hence we select keyframe which is located near the midpoint between one shot boundary and the next shot boundary [10]. There are different techniques proposed for video summarization. In earlier techniques uniform sampling or random selection was done to select key frame [11]. In color-based approaches key frames are selected on significant change in color histograms with certain threshold [12] [13].

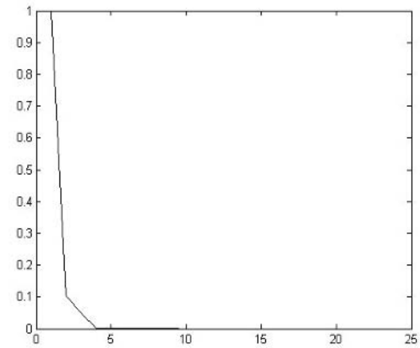


Fig. 5 Average of Distances between Previous and Current Cluster centers Versus the Iteration Steps [10]

Similarly some techniques are proposed with integration of low level and high level features to get desired key frames. Such as the frame having more skin color contents is more likely to be a selected key frame than other textures. Defaux [14] proposed a technique integrating motion and spatial analysis with skin color and face detection analysis. Recent techniques are based on clustering the similar features and selecting keyframe which is near to the centre of cluster [10]. Shot boundary based techniques of keyframe selection first divides a video sequence into shot and then from each shot key frame is selected. Different clustering algorithms are utilized amongst them most popular is k-means clustering algorithm. It reduces redundant candidate frames. Satoshi Hasebe [10] proposed a technique using wavelet feature vectors and applying k-means clustering algorithm. Fig 5 shows average of distances between previous and current cluster centers versus number of iterations. It narrates that it takes less than 5 iterations for the K-means clustering algorithm to converge to target centers. After 4rth iteration the centers are not further changed. This experiment is done on first 234 frames of video *Challenge at Glen Canyon* for 25 clusters. The whole algorithm is iterated for 25 times for a given initial sequence.

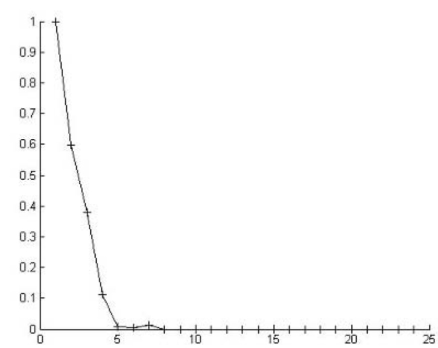


Fig. 6 Average of Distances between Previous and Current Cluster centers Versus the Iteration Steps

We have applied K-means clustering with the same initial sequence and same number of iterations on the first 234 frames of the same test sequence for 25 clusters. Fig.6 shows the results of clustering DTCWT features. It is shown that it took five iterations in order to converge to final cluster centers. So the proposed technique is comparable with the technique discussed earlier. The result of convergence heavily depends upon the initial cluster centers given to the k-means algorithm. We used same sequence for comparison.

V. VIDEO INDEXING

Advancement in digital video compression and broadcasting has been remarkably increased. This has opened new horizons in research. Video searching has its applications in digital libraries, teleconferences, surveillance systems and military defense systems. By video indexing we mean cataloging video segments with distinctive and searchable signatures. Where as querying video refers to the search for a similar video sequence in a database [3].

Process of shot boundary detection and key frame selection has greatly reduced the huge video data to digital image frames, whose quantity is equal to number of shots in the whole video sequence. Fig. 7 shows the whole framework of proposed video indexing system. Process of shot boundary detection and key frame feature selection is done for the whole video database for once. When a new video sequence is added in the database the whole process is repeated. This is offline process which is to be done for creating our feature database. In this feature database each feature corresponds to a key frame which is part of a single shot. In other words a feature is representing a shot. In online process (represented with white hollow arrows) a query shot is provided to the system. For this query shot process of feature extraction and key frame feature selection is performed and query features are obtained. This query feature representing the query shot is compared with all the features in the database. Different distance metrics are tried and correlation based distance metric performed well for the comparison of dual tree complex wavelet features. Results are the distances of query feature with the feature database. Now we sort the distances and indexes of the N sorted distances corresponds to key frames of the desired shots. These shots are selected and displayed to the user. There are different techniques to display results. Popular technique is to display thumbnails of video shots or display the key fames of the shots. We have displayed key frames of the resulting shots as shown in Figure 9. Different kind of characteristics of frames has been utilized for query systems. Color based features have been extensively used in content based image and video retrieval systems. Especially in video retrieval systems color histogram based techniques are invariant to geometrical operations. Consecutive key frames of a shot are quite similar to each other having little geometrical variations. Histogram differences are applied to get desired results [15]. Color based feature integrated with color layouts are desirable. Wavelets

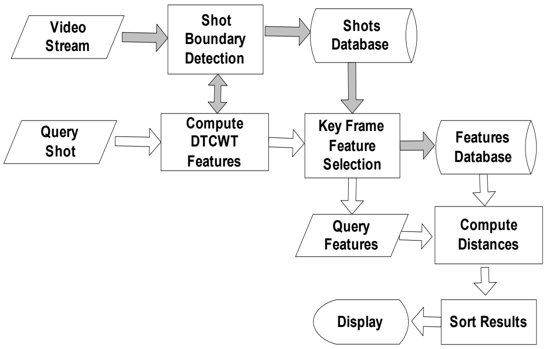


Fig. 7 CBVR System Overview

and Gabor filters are applied to the key frames to obtain texture features. Some techniques are proposed using the building blocks of textures to segment texture regions [16] [17]. Temporal analysis of a video sequence gives features specific to the video. Temporal analysis requires decomposing video into its basic elements [18]. In CBVR systems shot boundary detection is usually a pre-processing. Following are the steps of proposed algorithm:

Step1: For a shot select every frame and compute its dual tree complex wavelets transform features as described in section II.

Step2: Compute average of the features extracted in step1. Then select minimum distance of the features with the average. It provides feature of the frame which are more redundant and not the part of shot transitions. As shot transition frames are very few in number. For each shot select a single key frame which provide its best description. As explained in section IV.

Step3: Repeat step1 and step2 for each shot present in database. Create a feature database of key frames selected in step1 and step2.

Step4: Extract features for query shot as in step1 and step2 and select key frame to create a query feature.

Step5: Query feature is compared with each of the vector in feature database using following correlation based similarity metric.

$$d_{rs} = 1 - \frac{(x_r - \bar{x}_r)(x_s - \bar{x}_s)'}{\sqrt{\left[(x_r - \bar{x}_r)(x_r - \bar{x}_r)'\right]\left[(x_s - \bar{x}_s)(x_s - \bar{x}_s)'\right]}} \quad (2)$$

Step6: Select N frames with minimum distances. Shots corresponding to these key frame features are displayed as result.

VI. RESULTS

For the purpose of video indexing datasets with similar kind of shots grouped into different sets are required. In other words the database has known number of similar sets of shots. This makes possible to evaluate the precision and recalls graphs and comparison of different techniques. For this purpose we created our own video database with 7.2 Mega

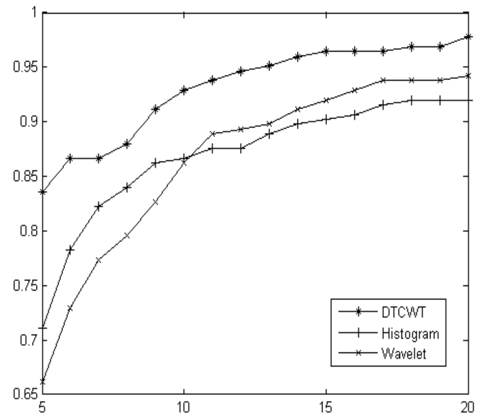


Fig. 8 Precision Rate versus Number of Shots Retrieved

pixels digital camera. A scene is captured from 5 different angels. Total 45 shots are captured for test purpose. Each shot contains on the average 245 frames. So total frames processed in our test are 45 x 245, which are nearly 11 thousand frames. Fig. 8 shows results of query retrieval; where on x-axis number of shot required and on y-axis precision of retrieved shots. **PRECISION** is defined as ratio of relevant records retrieved to the total records retrieved. **RECALL** is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. These are the performance measures of data retrieval. Results of DTCWT retrieval system is compared with classical color histogram based technique and wavelet based features [4] indexing. DTCWT performed better. We have already applied DTCWT indexing with support vector machines on images [19] and it produced

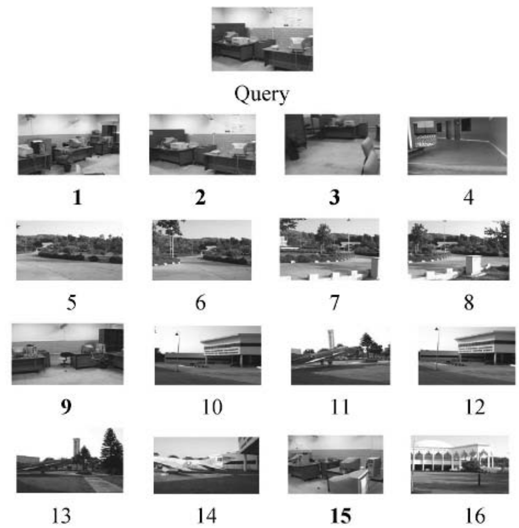


Fig 9 Retrieved Results of Proposed System

remarkable results. As shown in fig. 8, for retrieval of 5 shots on a query shot the proposed system retrieves 84% precise results. We have compared Euclidean, Correlation, Cosine and SEuclidean distance metrics for a given query. Results show that correlation distance metrics performs best. Figure 9 shows retrieval results for a test query shot. For this query sixteen relevant shots are demanded. Results shows that the retrieved shots quite similar to each other. Our database was organized as set of five similar shots. The figure shows that retrieved shots one, two, three, nine and fifteen belongs to similar set. So for this particular query system has produced 100% retrieval results at a recall of fifteen shots. This is about 33% of total shots. These results of query heavily depend upon selection of keyframes. If a keyframe describe the maximum contents of the shot the shot and it is compared with any other frame of the same shot, there is maximum possibility that the comparison result will be positive. Where as selection of a keyframe heavily depends upon correct shot boundary detection.

VII. CONCLUSION

In this paper we have proposed a technique for video indexing including key frame selection and shot boundary selection. We have used dual tree complex wavelet features for the entire framework. We have shown how correlation based distance measures perform better for the indexing from other distance measures. We have compared performance of our proposed system with the existing wavelet based and histogram based features. Our system provides satisfactory results.

ACKNOWLEDGMENT

We thank Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI) for providing very conducive research environment as well as financial and moral support.

REFERENCES

[1] Jung Hwan Oh, Quan Wen, Sae Hwang, Jeongkyu Lee, “Video Abstraction” *Book Chapter XIV*, The university of Texas at Arlington, USA

[2] Le Gall D, MPEG: A video compression standard for multimedia applications. *Commun. ACM*, 34(4): 46–58, 1991

[3] Xiadong Wen, Theodore D. Huffmire, Helen H. Hu, Adam Finkelstein “Wavelet-based video indexing and querying” *Multimedia Systems 7 Springer – Verlag*, pp. 350-358 1999

[4] Satoshi Hasebe, S. Muramatsu, S. Sasaki, J. Zhou and H. Kikuchi “Two-Step Algorithm for detecting Video Shot Boundaries in a Wavelet Transform domain ” *Proc. 4rd International Symposium on Image and signal Processing and analysis (ISAP03),* pp 245-250, Rome,2003

[5] Kingsbury, N.G.,”The Dual Tree Complex Wavelet Transform: a new efficient tool for image restoration and enhancement”, *Proc. European Signal Processing Conf.*, pp319-322, 1998.

[6] Peter, R. and Kingsbury, N.,”Complex Wavelets Features for Fast Texture Image retrieval”, *Proc. IEEE Int. Conf. on Image Processing,*, 1999.

[7] Chengcui Zhang1, Shu-Ching Chen1, Mei-Ling Shyu2, “PixSO: A System for Video Shot Detection” *ICICS*-PCM Singapore, 2003

[8] Yousri Abdeljaoued, Touradj Ebrahimi, Charilaos Christopoulos and Ignacio Mas Ivars, “A New Algorithm for Shot Boundary Detection” *EURASIP*, 2000

[9] Dong Zhang, Wei Qi, Hong Jiang Zhang, “A New Shot Boundary Detection Algorithm”,.*IEEE Pacific Rim Conference on Multimedia*, pp.63-70 , 2001

[10] Satoshi Hasebe, Makoto Nagumo, Shogo Muramatsu and Hisakazu Kikuchi, “Video Key Frame Selection by Clustering Wavelet Coefficients” , *EURASIP*, Austria, 2004

[11] Hammoud, R., & Mohr, R., “A probabilistic framework of selecting effective key frames from video browsing and indexing”,. *Proc. of International Workshop on Real-Time Image Sequence Analysis*, Oulu, Finland, 79-88, 2000

[12] Li, Y., Zhang, T., & Tretter, D, “An overview of video abstraction techniques”, Retrieved from the World Wide Web: http://www.hpl.hp.com/techreports/2001/HPL-2001-191.html, 2001

[13] Yu, H., & Wolf, W, “A visual search system for video and image databases”, *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Ottawa, Canada, 1997

[14] F. Dufaux, “Key frame selection to represent a video”, *ICME* , 2000.

[15] J Puzicha, Y. Rubner, C. Tomasi and J. M. Buhmann “Empirical Evaluation of Dissimilarity Measures of Color and Texture”, *Proc. of IEEE International Conference on Computer Vision (ICCV’99)*, 1999

[16] K. I. Chang, K. Bowyer and M. Sivagurunath, “Evaluation of texture segmentation algorithms”, *in proc. of the conference on computer vision and pattern recognition (CVPR’99*), volume 1, page 294-299, Fort Collins, Colorado, 1999

[17] N. Paragios and R. Deriche. “Geodesic active Contours of Texture Segmentation ”, *Technical report 3340, POBOVIS, INRIA*, Sophia – Antipolis, France, 1998

[18] Stephane marchand Maillet, “Content Based Video Retrieval-An Overview”, *Technical report on Vision*, Geneva University, Switzerland. 2000

[19] Adeel Mumtaz,SAM Gillani, Tahir Jameel, “A Novel Texture Image Retrieval System Based on Dual Tree Complex Wavelet Transform and Support Vector Machines”, in *proc. IEEE International Conference on Emerging Technologies*, Peshawar, Pakistan 2006

Separation of Shape and Data

Thomas Nitsche

Abstract—**In parallel and distributes systems we have to consider aspects of communication, synchronization and data movement besides the actual algorithmic solution. In order to enable the clear separation of these aspects [29] we have to separate the structure or shape of data structures from the actual data elements themselves. While the latter can be used to describe algorithmic aspects, the former enables us to derive the communication [27].**

In this paper we formalize the notion of shape and data elements and how they can be separated as well as used to reconstruct a valid data structure again for arbitrary container types, i.e. parameterized, nested algebraic data types and arrays. The *shape* function removes the data elements from the container type and replaces them by a dummy element * (or alternatively, their identifier). The *data* function extracts the list of data elements contained in the data structure; while a *re_cover* function can be used to reconstruct a data structure from its shape and a consistent list of data elements. All these functions can be characterized as special cases of a general traversal function operating on the data structure or its shape.

We have used this as the semantical basis for handling overlapping data distributions not only over arrays as in commonly used approaches but over arbitrary (container) types.

Index Terms—**Parallel Programming, Separation of Concerns, Data Types, Container Types, Shape of Data Structures.**

I. Introduction

Programming parallel and distributed systems requires programmers to consider the distribution of the work respectively the data onto the different processors, as well as synchronization and data exchange, i.e. communication, between them. For this reason the extraction of the data elements from a given data structure, their distribution and communication has to be handled in a parallel program. This can be done explicitly by the programmer in a low-level way as in MPI [31], by the system or combinations thereof.

The decision of which data will be communicated and when depends largely on the characteristics of the parallel computer such as its network topology, bandwidth, etc. To achieve maximum efficiency, many low-level machine- and algorithm-specific details have to be considered. The resulting parallel program is highly problem- and machine-specific, which makes it error-prone if programmed by explicit message-passing [12] and difficult to port to another parallel machine or to reuse the code for another program. Since one of the ideas behind grid computing is to make parallel computing power as easily available on the market as today's electricity or telephony services [11], parallel programs should not be written for a specific parallel machine but rather parameterized using certain architectural parameters. The number of available processors is one such architectural parameter; others are the computing power of each processor, its memory, the network parameter, etc.

In order to achieve a higher level of programming effectiveness and coordination of parallel activities, we can use algorithmic skeletons [3], [7], [8], [9], [30] as a kind of collective parallel operation instead of directly using low-level (point-to-point) communication operations. Well known examples for such generic operations include the *map* skeleton, which applies a function to all data elements, the *zip* skeleton, which combines two (or more) data structures, and the *reduce* skeleton, which combines all data elements to a "sum" using some binary operator:[1]

$$\mathrm{map}(f)([a_1, \ldots, a_N]) = [f(a_1), \ldots, f(a_N)] \quad (1)$$

$$\mathrm{zip}(\otimes)([a_1, \ldots, a_N], [b_1, \ldots, b_N]) = [a_1\otimes b_1, \ldots, a_N\otimes b_N] \quad (2)$$

$$\mathrm{reduce}(\oplus)([a_1, \ldots, a_N]) = a_1 \oplus \ldots \oplus a_N \quad (3)$$

In an array processing language this would correspond to $f(A)$, $A\otimes B$, and $\oplus(A)$ as used in, e.g., $max(abs(A) - B)$.

Operations on locally available data elements now correspond to purely local operations operating merely on the data elements themselves, while accesses to elements residing on other processors imply communication requirements, where the structure of the data type determines the communication structure. If we thus separated the data elements from the data structure (i.e. shape) itself, we could describe the parallel operations independently from the actual data types and use the shape for deriving the communication [27].

The remainder of this paper is organized as follows. Section II describes the concept of shape in an informal way, while section III gives its formal definition and derives related properties. Finally, section IV discusses related work and concludes.

II. Shape and Content of Data Structures

Every data structure has two aspects: its shape and its content (or data). Moreover, there is a mapping from shape to content. The content, for example, is the set of elements in the fields of a matrix, in the nodes of a tree, and so on, while the shape corresponds to the index range of a matrix or the node

Thomas Nitsche was with Technical University Berlin, Germany. He is now with the Research Institute for Communication, Information Processing and Ergonomics (FGAN/FKIE), Neuennahrer Straße 20, 53343 Wachtberg, Germany; e-mail: nitsche@fgan.de.

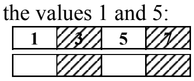
[1] The data elements need an order if the binary operator ⊕ is not associative and commutative.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 455–461.

structure of a tree. Separating shape from content allows us to handle the computational operations on the data elements independently of their mapping and ordering aspects.

Let us look at some examples. The simplest one is a single data element, e.g. a real value like π. Its shape is a placeholder for a real value, while its content corresponds to the value of the data element. Similarly, the shape of a tuple, e.g. a pair *&(3.1415, 2)* consisting of a real (*3.1.415*) and an integer (*2*) value, corresponds to the storage place for the values. Its content is the list of the data values. Because the values have different types (*real* and *int*, respectively), we must encode them in a list of the sum of the data types. Thus, formally, the content $\langle in_1(3.1415), in_2(2)\rangle$ is of type *list[real + int]*.

In these simple examples, the shape corresponds directly to the type of the data.[2] The same still holds for arrays. However, in this case it is also sufficient to use the size of the array together with the type of the data elements as shape information. MPI data types allow strides in arrays, meaning that not all array elements are used but only those in a regular interval. In the example below, the shape only contains the elements with even index, while the elements with odd index are omitted. Thus the content only consists of the values 1 and 5:

Array with stride 2:
Shape:


Content: $\langle 1, 5\rangle$

For nested, inhomogeneous arrays, the shape is merely the nested array structure. The list of data elements can be obtained by flattening [20] the nested arrays, i.e. by putting all data elements into a single list. For algebraic data types, we have to consider the structure, i.e. the graph of their cells, as the shape. In the case of linear lists, this is merely the sequence of cons cells, while for trees the traversal order of the elements within the data structure determines the order of the data elements within the content.

A. Shape Theory

Shape theory [18], [16] formally describes the separation of shape and data and how data is stored within data structures. Shapely types can be formally described as a pullback in the sense of category theory.

$$\begin{tikzcd} \text{Object[Data]} \arrow[r, "data"] \arrow[d, "shape"'] & \text{list[Data]} \arrow[d, "\#"] \\ \text{Shape} \arrow[r, "arity"'] & N \end{tikzcd} \tag{4}$$

The semantics of (4) are as follows. Given two morphisms *arity: Shape* → *N* and *#: list[Data]* → *N* (where # computes the length of a list), then there exists an – up to isomorphisms – uniquely determined object *Object[Data]* with morphisms *shape: Object[Data]* → *Shape* and *data: Object[Data]* → *list[Data]* such that the above diagram commutes, i.e. it satisfies $arity \circ shape = \# \circ data$. This means that the data structure *Object[Data]* can be constructed from its shape and the list of its data elements.[3]

B. Representing Shapes

In the language FISh (Functional + Imperative = Shape) [17] and its predecessor Vec the shape of a data structure is merely its size, i.e. the number of data elements, together with the shape of the elements. This allows the description of nested homogeneous vectors, all subvectors being required to have the same shape and hence the same length. The shape information thus describes the memory consumption of a certain element because here the shape corresponds to the size (required bytes) of a data element in the case of basic data types like *bool*, *nat* or *real*, and otherwise to the size of a vector (and the shape, i.e. the size of its components.) Shape analysis can be used for memory management and program optimization.

This is similar to the language SAC (Single Assignment C) [14]. It offers multidimensional, homogeneous arrays and – as in APL – dimension-invariant array operations [13]. Arrays in SAC consist of two parts: a data vector containing the data values, and a separate shape vector holding the array sizes in the different dimensions. Access to the shape vector is allowed within the language SAC. Consider, for example, a three-dimensional 2×2×3 matrix *A* with data elements 1, …,12. It is defined as *A = reshape([2,2,3], [1,2,3,4,5,6,7,8,9,10,11,12])*. Then, *dim(A)* returns the number of dimensions of the matrix *A*, i.e. the value 3, while *shape(A)* yields the shape vector *[2, 2, 3]* with the sizes within each dimension. This shape vector can be used to generate other shapes and hence new arrays, e.g. *genarray(min(shape(A), shape(B)))* computes the minimal size within each dimension of matrices *A* and *B*. Note, however, that only homogeneous vectors are allowed in SAC, i.e. all subvectors of a nested matrix must be of the same size.

In Nesl [4] and Nepal [5], [23], the subvectors of a nested array may have different lengths, thus allowing us to model not only dense but also sparse matrices. Internally, the nested vectors are subject to the flattening transformation and are represented by a data vector and a shape vector containing the sizes of the different subvectors [20]. For example, the nested vector [[1,2,3,4,5], [], [6], [7,8,9,10], [11,12]] is represented by the data vector [1,2,3,4,5,6,7,8,9,10,11,12] and the size vector [5,0,1,4,2], which contains the sizes of each sublist of the data.

ZPL [6] offers the concepts of regions that are abstract index sets with no associated data and grids as abstractions of processor sets [10]. Since ZPL is an array processing language, regions are based on arrays and indexes.

C. Shapes of Parameterized Algebraic Data Types

We do not want to deal only with arrays or lists but also with arbitrary algebraic data types, so the size or the index

[2] We will later modify the notion of shape slightly. Instead of storing a dummy value of the corresponding data type, we replace the data values by $* \in 1 = \{*\}$ (cf. Def. 3).

[3] Note that the morphisms have to take into account the traversal order within the data structure. Otherwise, the elements can be arbitrarily re-ordered within *Object[Data]* because permutations are isomorphisms on lists.

vector is not sufficient in our case. Instead, we have to pay attention to how elements are stored within a data structure. Formalizing our notion of shapes, we allow arbitrary container types, i.e. arbitrary (nested) combinations of parameterized algebraic data types and arrays.

Definition 1 (Container Type). The set of *container types* $C(V,B)$ over sets of base types B and type variables V is defined as

$C(V,B) \rightarrow$	B	Base type
$\mid$	V	Type variable
$\mid$	$C(V,B) \times \ldots \times C(V,B)$	Product
$\mid$	$C(V,B) + \ldots + C(V,B)$	Sum ∇

A container type $t[V] \in C(V,B)$ is an algebraic data type parameterized with the set of type variables V. $\pi_j\colon t_1 \times \ldots \times t_k \rightarrow t_j$ denotes the projection of the j-th component, while $in_j\colon t_j \rightarrow t_1 + \ldots + t_k$ is the usual injection function. The final object, i.e. the one-elementary set $1=\{*\}$, corresponds to the empty product, while the initial object $\varnothing$ corresponds to the empty sum. Since arrays play an important role in parallel programming, we should treat them explicitly. However, they are just special cases of product *types* $C(V,B)^n$, so we can omit them in the definition of shape and data functions. Base types and arguments for the type parameter variables can be arbitrary types including function types. $C(\varnothing,B)$ is the usual set of (unparameterized) algebraic data types together with arrays. Abbot et al. [1] give a more general, categorical definition of containers. In the case of locally cartesian closed categories, this is equivalent to the notion of shapely types [18].

The idea behind the concept of container types is that the values of such a parameterized type represent the data structure, while the type variables serve as holes or placeholders for the data elements stored within the data structure.

Examples for container types include tuples like $pair[\alpha,\beta] \in C(\{\alpha,\beta\}, \varnothing)$, linear lists $list[\alpha] \in C(\{\alpha\}, \varnothing)$, and (non-empty) search trees $tree[\alpha] \in C(\{\alpha\}, \{int\})$, that store data elements of type α according to a key element of type *int*:

$$pair[\alpha,\beta] = \alpha \times \beta \quad (4)$$

$$list[\alpha] = 1 + \alpha \times list[\alpha] \quad (5)$$

$$tree[\alpha] = \alpha + tree[\alpha] \times int \times tree[\alpha] \quad (6)$$

Thus the structure of a container type $C(V,B)$ represents the shape, while the parameters of the type variables V contain the data elements. The elements of the base types contain additional information about the data structure, i.e. its shape. This holds, e.g., for the search tree, where the key elements are part of the shape.[4]

III. Separating Shape and Data

A. Definition of Shape

Proposition 2 (Shape Function). Let $t[V] \in C(V,B)$ be a parameterized container type over parameter type variables $V=\{V_1,\ldots,V_m\}$, $t[P]$ be an actualization of type $t[V]$ with parameter types $P=\{P_1,\ldots,P_m\}$, $f_P\colon t[V] \rightarrow t[P]$ be a parameter function defined as a lifting of $f_P(v_i)=p_i$ (for $v_i \in V_j \in V$, $p_i \in P_j \in P$) from $V \rightarrow X$ to $t[V] \rightarrow t[P]$, and, finally, $1\colon t[V] \rightarrow t[1]$ be the lifted function from $V \rightarrow 1$ (with $1=1^m$) to $t[V] \rightarrow t[1]$ defined as $1(v_i)=*$ for $v_i\colon V_j \in V$. Then there exists a uniquely determined function $shape\colon t[P] \rightarrow t[1]$, such that $shape \circ f_P = 1$, i.e. the diagram in Fig. 1 commutes. □

$$\begin{tikzcd} t[V] \arrow[r, "f_P"] \arrow[dr, "1"'] & t[P] \arrow[d, "shape"] \\ & t[1] \end{tikzcd}$$

Fig. 1. Definition of the shape function

Proof. We define the function $shape_{t[P]}$ as $shape_{Vj}(p_i)=*$ for $b_i \in B_j \in B$, and as input preserving in all other cases (cf. Fig. 2). Because $shape_{Vj}(f_p(p_i))=shape_{Vj}(p_i)=*=1(v_i)$, and the functions are structurally equivalent in all other cases, it follows directly that $shape \circ f_P=1$. If $s\colon t[P] \rightarrow t[1]$ is another function satisfying $s \circ f_P = 1$, structural induction using case distinction of the type $t[P]$ shows $s = shape$ and hence its uniqueness. □

The function *shape* removes the data elements from a given data structure and leaves the structure – with the data replaced by * – alone. This can be expressed (cf. [28]) as

$$shape = map((\lambda v_j\colon P_j.*)_{j=1,\ldots,m}) \quad (7)$$

We call the resulting structure shape. The (omitted) data elements can be extracted using the function *data*. For example, for the data structure $d = [[1,2], [], [3,4,5], 6]$ containing a nested array, it holds:

- o shape(d) = [[*,*], [], [*,*,*], *]
- o data(d) = ⟨1,2,3,4,5,6⟩

Definition 3. Let $t[V] \in C(V,B)$ be a parameterized container type over parameter type *variables* $V=\{V_1,\ldots,V_m\}$ and $t[P]$ be an actualization of type $t[V]$ with parameter types $P=\{P_1,\ldots,P_m\}$.

1) For any data structure $d \in t[P]$, the structure $shape(d)$ is called **shape** of d.
2) The **size** of the shape is the number of * elements (as replacements or placeholders for data elements) within it, i.e. $arity\colon t[1] \rightarrow N$ is defined as in Fig. 2.
3) The list of **data elements** within a data structure $d \in t[P]$ can be retrieved via the data extraction function $data\colon t[P] \rightarrow list[P_1+\ldots+P_m]$, which is defined as lifting of $data(p_i)=\langle in_j(p_i)\rangle$ for $p_i \in P_j$.
4) A shape $s \in t[1]$ together with a list of data elements $\langle d_1,\ldots,d_{arity(s)}\rangle \in list[P_1+\ldots+P_m]$ can be used to generate a data structure by the (partial) function ***re_cover***: $t[1] \times list[P_1+\ldots+P_m] \rightarrow t[P]$, which is defined as an extension of $re\_cover(*, \langle in_j(p_i)\rangle) = p_i$. ∇

For example, the shape of a pair $d = \&(3.1415, 2) \in$

[4] If we defined the tree as $tree[\alpha,\beta] = \alpha + tree[\alpha] \times \beta \times tree[\alpha]$, the key elements would have been part of the data and not of the shape as in (6).

Let $b_i \in B_j \in B$, $p_i \in P_j \in P$, $V_j \in V$, $x, x_i \in t[P]$, $s, s_i \in t[1]$, $ds, d_i \in list[P_1+\ldots+P_m]$:

$$\begin{array}{lll}
shape_{Bj}(b_i) & = b_i & \text{(base type)} \\
shape_{Vj}(p_i) & = * & \text{(data value)} \\
shape_{T1\times\ldots\times Tk}((x_1,\ldots, x_k)) & = (shape_{T1}(x_1),\ldots, shape_{Tk}(x_k)) & \text{(product)} \\
shape_{T1+\ldots+ Tk}(in_j(x)) & = in_j(shape_{Tj}(x)) & \text{(sum)}
\end{array}$$

$$\begin{array}{llll}
data(b_i) & = \langle\rangle & arity(b_i) & = 0 \\
data(p_i) & = \langle in_j(p_i)\rangle & arity(*) & = 1 \\
data((x_1,\ldots,x_k)) & = data(x_1) +\!\!\!+ \ldots +\!\!\!+ data(x_k) & arity((s_1,\ldots,s_k)) & = \sum_{i=1}^{k} arity(s_i) \\
data(in_j(x)) & = data(x) & arity(in_j(s)) & = arity(s)
\end{array}$$

$$\begin{array}{ll}
re_cover(b_i, \langle\rangle) & = b_i \\
re_cover(*, in_j(p_i)) & = p_i \\
re_cover((s_1,\ldots,s_k), & = (y_1,\ldots,y_k) \\
\qquad \langle d_1,\ldots,d_l\rangle) & \quad \text{where } y_i = re_cover(s_i, \langle d_{\sum_{j=1}^{i-1} arity(sj)+1},\ldots, d_{\sum_{j=1}^{i} arity(sj)}\rangle) \\
re_cover(in_j(s), ds) & = in_j(re_cover(s, ds))
\end{array}$$

Fig. 2. Functions shape, data, arity and re cover

pair[real, int] is *shape(d) = (*,*)* $\in$ *pair[1,1]*, while its list of data elements is $data(d)=\langle in_1(3.1415), in_2(2)\rangle \in list[real+ int]$, where in_j encodes the corresponding data type, i.e. *real* or *int*.

Analogously, the shape of the two-element list of natural numbers $\langle 1,2\rangle$ and that of the list of lists $\langle\langle 1,2\rangle, \langle 3,4,5,6\rangle\rangle$ is merely a list $\langle *,*\rangle$ with two * elements. It is obtained by removing the data elements, i.e. the natural numbers or the sublists, and replacing them by *. The arity (size) of the shape, i.e. the number of (removed) data elements, is equivalent to the length 2 of the list. The list of data values is, in our example, the original list itself, so the function *data* (see Def. 3) is the identity here.[5]

B. Special Cases: Shape of Data Lists and Data of Shapes

Note that the type of the shapes *t[1]* is merely a special case of an actualization of parameterized container types with parameter types *1*. The function *data* is therefore also defined on shapes. However, since all data elements correspond to the single value *, the values in the result of *data(s)* for a shape $s \in t[1]$ are of no interest. But the function yields the type information of the former data elements:

$$data: t[\{1,\ldots,1\}] \rightarrow list[1+\ldots+1]^6$$

Thus for $data(s) = \langle e_1,\ldots, e_{n-1}\rangle$, it holds $e_i = in_j(*)$ for some $j \in \{1,\ldots,m\}$. This gives us the information that the *i*-th data element in the data structure $d \in t[\{P_1,\ldots,P_m\}]$ with $shape(d)=s$ is of type P_j.

Since the type $list[P_1+\ldots+P_m]$ is also merely a special case of an actualization of a parameterized data type $list[V_1+\ldots+V_m]$ with *m* parameter type variables $V_1,\ldots, V_m$, the function *shape* is also defined here.

C. Consistency Conditions

The application *re_cover(s, ds)* of the partial function *re_cover* is only defined if it is provided with enough data values to replace all placeholder * elements in the shape *s* with corresponding data elements from *ds*. This is expressed by the condition $\#(ds) \geq arity(s)$. Furthermore, the data elements of

$$\begin{tikzcd}
t[P] \arrow[r, "f"] \arrow[dr, "shape"'] & t[P'] \arrow[d, "shape"] \\
& t[1]
\end{tikzcd}$$

Fig. 3. Definition of shape-preserving functions

the list *ds* must have the proper type to rebuild a correctly typed data structure. With the previously defined functions, we can express these conditions as

$$shape(ds) = data(s) \qquad (8)$$

Definition 4 (Consistency). Let $ds \in list[P_1+\ldots+P_m]$ be a list of data elements.

1) *ds* is called **consistent** with a shape $s \in t[1]$ iff *shape(ds) = data(s)*.
2) *ds* is called consistent with a data structure $d \in t[P]$ iff *ds* is consistent with the shape of *d*, i.e. *shape(ds) = data(shape(d))*. ∇

Consistent data lists can be used to reconstruct a correctly typed data structure from a shape structure (Def. 4-(1)), or they can be used to replace all data elements in a data structure, yielding a correctly typed new data structure (Def. 4-(2)). If a function on lists of data elements operates in such a way that it keeps this type information, we call this function **type-preserving** or, in the case of arbitrary data types, **shape-preserving** (cf. Fig. 3).

Definition 5. $f: t[P] \rightarrow t[P']$ is called **shape-preserving** iff $shape = shape \circ f$. ∇

D. General Traversal Function

The function *data* is defined by an in-order traversal of the data structure. This can be expressed by a general traversal operation (see [24]) on the data structure.

Definition 6 (General Traversal). Let $t \in C(V,B)$ be a parameterized container type with $V=\{V_1,\ldots,V_m\}$. Further, let $\alpha=\{\alpha_1,\ldots,\alpha_m\}$ and $\beta=\{\beta_1,\ldots,\beta_m\}$. Then a general traversal

$$traverse: (\alpha_1\times\gamma\rightarrow\beta_1\times\gamma) \times \ldots\times (\alpha_m\times\gamma\rightarrow\beta_m\times\gamma)\ t[\alpha]\times\gamma\rightarrow t[\beta]\times\gamma$$

on type *t* with a set of functions $f = (f_k)_{k=1,\ldots,m}$ can be defined as follows:

$$\begin{array}{lll}
traverse(f)(b_i, p) & = (b_i, p) & b_i{:}B_j \in B,\ p{:}\gamma \\
traverse(f)(v_i, p) & = f_j(v_i, p) & v_i{:}\alpha_j,\ p{:}\gamma \\
traverse(f)((x_1,\ldots,x_k), p) & = ((y_1,\ldots,y_k), r_n) & \\
 & \quad \text{where } r_0=p & p{:}\gamma \\
 & \qquad (y_i, r_i) = traverse(f)(x_i, r_{i-1}) & i=1,\ldots,k \\
traverse(f)(in_j(x), p) & = (in_j(y), r) & p{:}\gamma \\
 & \quad \text{where } (y, r) = traverse(f)(x, p) &
\end{array}$$

∇

We can easily prove that the **traverse-fusion law** is satisfied, i.e. it holds $traverse(g \circ f) = traverse(g) \circ traverse(f)$ and $traverse(id_{\alpha\times\beta}) = id_{t[\alpha]\times\beta}$.

[5] This is due to the fact that the list $list[P_1]$ is only parameterized over one data type P_1, so $data(p) = \langle in_1(p)\rangle \equiv \langle p\rangle$ because $P_1 \equiv +_{i=1}^{1} P_i$ (see Def. 3-(3)).

[6] where 1 appears m times each

The functions defined in Def. 3 can now be described as special cases of this general *traverse* operation operating on a data structure or its shape.

Proposition 7 (Special Cases of General Traversal).

1) $\langle \text{shape}, \text{data}\rangle = \text{traverse}((\lambda v{:}P_j, ds.\ (*, ds \mathbin{+\!\!+} \langle in_j(v)\rangle))_{j=1,\ldots,m}) \circ (\lambda d.(d, \langle\rangle))$
2) $\text{arity} = \pi_2 \circ \text{traverse}((\lambda v,n.\ (v, n+1))_{j=1,\ldots,m}) \circ (\lambda s.(s, 0))$
3) $\text{re\_cover} = \pi_1 \circ \text{traverse}((\lambda v{:}P_j, \langle in_j(d_1)\rangle \mathbin{+\!\!+} ds'.(d_1, ds'))_{j=1,\ldots,m})$

$\square$

Proof (by structural induction). We will sketch the proof for *shape* and *data* (Prop. 7-(1)). The complete proof can be found in [28], Th. 4.18.

Let $f = (\lambda v{:}P_j, ds.\ (*, ds \mathbin{+\!\!+} \langle in_j(v)\rangle))_{j=1,\ldots,m}$. Then the following holds:

(base types): $b \in B_j \in B$

$(\text{shape}(b), \text{data}(b)) = (b, \text{data}(b)) = (b, \langle\rangle) = \text{traverse}(f)(b, \langle\rangle)$

(data elements): $p \in P_j \in P$

$(\text{shape}(p), \text{data}(p)) = (*, \text{data}(p)) = (*, \langle in_j(p)\rangle) = f_j(p, \langle\rangle) = \text{traverse}(f)(p, \langle\rangle)$

(sum): $x \in t[P]$

$$\begin{aligned}
&(\text{shape}(in_j(x)), \text{data}(in_j(x))) \\
&= (in_j(\text{shape}(x)), \text{data}(in_j(x))) && \text{(def. shape)} \\
&= (in_j(\text{shape}(x)), \text{data}(x)) && \text{(def. data)} \\
&= (in_j(y), r) \text{ where } (y,r) = (\text{shape}(x), \text{data}(x)) && \text{(funct. appl)} \\
&= (in_j(y), r) \text{ where } (y,r) = \text{traverse}(f)(x, \langle\rangle) && \text{(ind. prec.)} \\
&= \text{traverse}(f)(in_j(x), \langle\rangle) && \text{(def. traverse)}
\end{aligned}$$

(product): The case for products is demonstrated analogously (with the help of a small lemma proving $traverse(f)(s,ds) = \langle id, \lambda ds'.ds \mathbin{+\!\!+} ds'\rangle \circ traverse(f)(s, \langle\rangle)$).

Thus the assertion follows by structural induction.

$\square$

E. Inverse of Shape of Data

Traversal orderings of the data structure other than the one defined in Def. 6 are possible as well, provided the relationship between a certain data element and its former position within the data structure is defined. In our case, this property holds for the data extraction function *data* and its reverse operation *re_cover*:

Theorem 8 (Inverse of shape and data).

1) $\text{re\_cover} \circ (\text{shape} \times \text{data}) = id_{t[P]}$
2) $(\langle \text{shape}, \text{data}\rangle \circ \text{re\_cover})(s, ds) = (s, ds)$, if ds consistent with s. $\square$

Proof. Follows as a corollary from Prop. 7 and the traverse-fusion law. $\square$

Due to Prop. 7 it is thus sufficient to define a traversal on a corresponding data type. This can then be used to derive implementations for extracting the data elements and the shape as well its reverse *re_cover* operation. We use this for handling overlapping data distributions with arbitrary container types, i.e. nested algebraic data types (see [27], [28] for details).

F. Consistency of Shape and Data

Since the number of placeholders for data elements within the shape corresponds to the number of *data* elements extracted from the data structure, the diagram in Fig. 5 (as a

$$\begin{tikzcd}
t(\{P_1,\ldots,P_m\}] \arrow[r, "data"] \arrow[d, "shape"'] & list[P_1+\ldots+P_m] \arrow[d, "shape"] \\
t[\{1,\ldots,1\}] \arrow[r, "data"'] & list[1+\ldots+1]
\end{tikzcd}$$

Fig. 4. Consistency between shape and extracted data elements

$$\begin{tikzcd}
t(\{P_1,\ldots,P_m\}] \arrow[r, "data"] \arrow[d, "shape"'] & list[P_1+\ldots+P_m] \arrow[d, "\#"] \\
t[\{1,\ldots,1\}] \arrow[r, "arity"'] & N
\end{tikzcd}$$

Fig. 5. Relationship between shape and data elements for container types

specialication of (4)) commutes (cf. Cor. 10). Moreover, not only the number of data elements extracted from a shape matches the size or arity of a shape, but even the types of the elements are such that we can reconstruct the data structure from the elements (and their shape).

Proposition 9. *data(d)* is consistent with *shape(d)*. $\square$

Proof. (base types): $b_i \in B_j \in B$

$\text{data}(\text{shape}(b_i)) = \text{data}(b_i) = \langle\rangle = \text{shape}(\langle\rangle) = \text{shape}(\text{data}(b_i))$

(data elements): $p_i \in P_j \in P$

$\text{data}(\text{shape}(p_i)) = \text{data}(*) = \langle in_j(*)\rangle = \text{shape}(\langle in_j(*)\rangle) = \text{shape}(\text{data}(p_i))$

(product): $\text{data}(\text{shape}((x_1, x_2))) = \text{data}((\text{shape}(x_1), \text{shape}(x_2))) = \text{data}(\text{shape}(x_1)) \mathbin{+\!\!+} \text{data}(\text{shape}(x_2)) = \text{shape}(\text{data}(x_1)) \mathbin{+\!\!+} \text{shape}(\text{data}(x_2)) = \text{shape}(\text{data}(x_1) \mathbin{+\!\!+} \text{data}(x_2)) = \text{shape}(\text{data}((x_1, x_1)))$

(sum): $\text{data}(\text{shape}(in_j(x))) = \text{data}(in_j(\text{shape}(x))) = \text{data}(\text{shape}(x)) = \text{shape}(\text{data}(x)) = \text{shape}(\text{data}(in_j(x)))$

Thus the assertion follows by structural induction.

$\square$

According to Def. 4, this means *data* $\circ$ *shape* = *shape* $\circ$ *data*, i.e. the diagram in Fig. 4 commutes. The data elements selected from a data structure are therefore obviously consistent with the data structure itself because they are of the corresponding number and types.

Corollary 10. $\text{arity} \circ \text{shape} = \# \circ \text{data}$ $\square$

Proof. For $d \in t[P]$, it holds $\text{arity}(\text{shape}(d)) = \#(\text{data}(\text{shape}(d))) = \#(\text{shape}(\text{data}(d))) = \#(\text{data}(d))$ (see Fig. 5). $\square$

Since $shape_{t[1] \to t[1]} = id_{t[1]}$ and $data_{list[P1+\ldots+Pm] \to list[P1+\ldots+Pm]} = id_{list[P1+\ldots+Pm]}$, the functions are idempotent:

$$\text{shape} \circ \text{shape} = \text{shape}$$

(9)

$$\text{data} \circ \text{data} = \text{data} \quad (10)$$

IV. CONCLUSION

Examples for exploiting shape in parallel programming are diverse. They include, among others, the data field model for indexed collections of data [15], PEI [32], or the FISh language [17]. The shape of data structures, or more specifically the sizes and indices of distributed arrays and matrices, are also handled in the (internal implementation of) the skeletons of the Muesli library [22].

Beckmann and Kelly adopt an interesting approach in optimizing regular matrix computations. They optimize the parallel execution of library calls like that of the Basic Linear Algebra Subroutines (BLAS) at runtime. Their basic idea is to suspend the actual execution of array operations, and optimize the combination of (hopefully) all library calls of the program. Since the arguments of library calls are known at runtime, they know the exact sizes of each matrix and how the output of one library call is used in later calls. This allows them to distribute the data in a way that might be non-optimal for one specific library call, but achieves better overall performance because additional data redistributions can be avoided [2], [21].

However, common approaches only work for one specific data type: the languages HPF, SAC [14], ZPL [6] or FISH [17], [16] operate on arrays or matrices, while NESL [4] and Nepal [5] use (nested) vectors. Other data types have to be encoded explicitly (by the user) [19].

In this paper we formalized the notion of shape and data elements and how they can be separated as well as used to reconstruct a valid data structure again. The approach works for arbitrary container types, i.e. parameterized algebraic data types, arrays and arbitrary nested combinations thereof. The *shape* function removes the data elements from the container type and replaces them by a dummy element * (or alternatively, their identifier). The *data* function extracts the list of data elements contained in the data structure, while a *re_cover* function can be used to reconstruct a data structure from its shape and a consistent list of data elements. All these functions can be characterized as special cases of a general *traversal* function operating on the data structure or its shape.

This allows the data elements and the shape to be processed independently from each other. We can especially distribute the data onto different processors, with the shape encoding the communication structure between them. We have used this as the basis to handle overlapping data distributions [28].

Moreover, this programming oriented separation serves as the basis for formally defining the notion of shapely functions, i.e. functions that change their shape only in a data-independent manner [25], [18]. Since here the shape can be calculated without having to know data from other processors, this allows precalculated communication schedules as well as security checks in distributed environments [26].

REFERENCES

[1] M. Abbott, T. Altenkirch, and N. Ghani, "Categories of containers", in *Proc. FOSSACS'03*, vol. 2620 of LNCS, Springer, 2003, pp. 23–38.

[2] O. Beckmann, "Interprocedural optimisation of regular parallel computations at runtime", PhD thesis, Imperial College, London, Jan. 2001.

[3] A. Benoit, M, Cole, S. Gilmore, and J. Hillston. "Flexible skeletal programming with eSkel", in *Proc. Euro-Par'05 Parallel Processing*, vol. 3648 of LNCS, Springer, 2005, pp. 761–770.

[4] G. E. Blelloch, "Nesl: A nested data-parallel language (3.1)", Carnegie Mellon Univ., Tech. Rep. CMU-CS-95-170, Sept. 1995.

[5] M. M. T. Chakravarty, G. Keller, R. Leshchinskiy, and W. Pfannenstiel, "Nepal - nested data parallelism in Haskell", in *Proc. Euro-Par'01 Parallel Processing*, vol. 2150 of LNCS, Springer, 2001, pp. 524–534.

[6] B. L. Chamberlain, S.-E. Choi, S. J. Deitz, and L. Snyder, "The high-level parallel language ZPL improves productivity and performance", in *Proc. IEEE International Workshop on Productivity and Performance in High-End Computing*, 2004.

[7] M. I. Cole, *Algorithmic Skeletons: Structured Management of Parallel Computation*, MIT Press, 1989.

[8] M. I. Cole, "Bringing skeletons out of the closet: a pragmatic manifesto for skeletal parallel programming", *Parallel Computing*, vol. 30, no. 3, pp.389–406, 2004.

[9] J. Darlington, Y. Guo, H.W. To, and J. Yang, "Functional skeletons for parallel coordination", in *Proc. Euro-Par'95*, vol. 966 of LNCS, 1995.

[10] S. J. Deitz, B. L. Chamberlain, and L. Snyder, "Abstractions for dnamic data distribution", in *Proc. IEEE Workshop on High-Level Parallel Programming Models and Supportive Environments*, 2004.

[11] I. Foster and C. Kesselmann, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, 1999.

[12] S. Gorlatch, "Send-receive considered harmful", *ACM Transactions on Programming Languages and Systems*, vol. 26, no. 1, pp. 47–56, 2004.

[13] C. Grelck and S.-B. Scholz, "SAC – from high-level programming with arrays to efficient parallel execution", *Parallel Processing Letters*, vol. 13, no. 3, pp. 401–412, 2003.

[14] C. Grelck and S.-B. Scholz, "SAC – a functional array language for efficient multi-threaded execution", *International Journal of Parallel Programming*, vol. 34, no. 4, pp. 383–427, Aug. 2006.

[15] J. Holmerin and B. Lisper, "Development of parallel algorithms in Data Field Haskell", *Proc. Euro-Par'00*, vol. 1900 of LNCS, Springer, 2000.

[16] C. B. Jay, "Costing parallel programs as a function of shapes", *Science of Computer Programming,* vol. 37, no. 1–3, pp. 207–224, May 2000.

[17] C. B. Jay and P. A. Steckler, "The functional imperative: shape!", in *Proc. ESOP'98*, vol. 1381 of LNCS, Springer, 1998, pp. 139–53.

[18] C.B. Jay, "A semantics for shape", *Science of. Computer Programm*ing, vol. 25, no. 2-3, pp. 251–283, 1995.

[19] G. Keller and M. M. T. Chakravarty, "Flattening trees", in *Proc. Euro-Par'98 Parallel Processing*, vol. 1470 of LNCS, 1998, pp. 709-719.

[20] G. Keller and M. Simons, "A calculational approach to nested data parallelism in functional languages", in *Proc. Asian Computing Science Conf.*, vol. 1179 of LNCS, Springer, Dec. 1996, pp. 234–243.

[21] P. H. J. Kelly and O. Beckmann, "Generative and adaptive methods in performance programming", *Parallel Processing Letters*, vol. 15, no. 3, pp. 239–256, Sep. 2005.

[22] H. Kuchen, "A skeleton library", in *Proc. Euro-Par'02 Parallel Processing,* vol. 2400 of LNCS, Springer, Aug. 2002, pp. 620–629.

[23] R. Leshchinskiy, "Higher-order nested data parallelism: semantics and implementation", PhD thesis, Technical Univ. of Berlin, Dec. 2005.

[24] E. Moggi, G. Belle, and C. B. Jay, "Monads, shapely functors and traversals", in *Proc. 8th Conf. on Category Theory and Computer Science (CTCS'99),* vol. 24 of Electronic LNCS, Elsevier, 1999, pp. 265–286.

[25] T. Nitsche, "Shapeliness analysis of functional programs with algebraic data types", *Science of Computer Programming*, pp. 225–252, 2000.

[26] T. Nitsche, "Secure communication in distributed environments – Ensuring consistency of communication structures by shapeliness analysis", presented at the Spring School on Security, Marseille, April 2005.

[27] T. Nitsche, "Deriving and scheduling communication operations for generic skeleton implementations", *Parallel Processing Letters*, vol. 15, no. 3, pp. 337–352, Sep. 2005.

[28] T. Nitsche, "Data distribution and communication management for parallel systems", PhD thesis, Dept. of Comp. Sci. and Electr. Eng., Technical Univ. of Berlin, Dec. 2005.

This page intentionally blank

[29] T. Nitsche, "Coordinating computation with communication", in *Proc. COORDINATION 2006*, vol. 4038 of LNCS, Springer, Jun. 2006, pp. 212–227.

[30] F. A. Rabhi and S. Gorlatch, Ed., *Patterns and Skeletons for Parallel and Distributed Computing*, Springer, 2003.

[31] M. Snir, S. W. Otto, S. Huss-Ledermann, D. W. Walker, and J. Dongarra, *MPI – The Complete Reference*, MIT Press, 1998.

[32] E. Violard, "Typechecking of PEI expressions", in *Proc. Euro-Par'97 Parallel Processing*, vol. 1300 of LNCS, Springer, 1997, pp. 521–529.

A Negotiation Model for Collaborative Decision Making in Large-Scale Multi-Agent Systems

Tom Wanyama
Department of Networks and Software Engineering
Faculty of Computing and Information Technology
Makerere University, P. O. Box 7062
Kampala, Uganda
wanyama@cit.mak.ac.ug

Abstract - Modeling agent negotiation is of key importance in building multi-agent system, because negotiation is one of the most important types of agent interaction. Negotiation provides the basis for managing the expectations of the individual negotiating agents, and it enables selecting solutions that satisfy all the agents as much as possible. Thus far, most negotiation models have serious limitation and weakness when employed in large-scale multi-agent systems. Yet, large-scale multi-agent systems find their use in major domains of human development such as space exploration, military technology, disaster response systems, and health technology. This paper presents an agent negotiation model which extends the capabilities of the model associated with the Agent Negotiation Engine for Collaborative Decision Making, to address the negotiation issues associated with large-scale multi-agents systems. The model utilizes Qualitative Reasoning and Game Theory algorithms to track the negotiation process, and a similarity criteria algorithm to manage the large amount of negotiation information associated with large-scale multi-agent systems. For completeness sake, the paper also presents the negotiation models from which the negotiation model for large-scale multi-agent systems evolved, as well as how and why the modifications were made.

I. INTRODUCTION

Negotiation is a form of agent interaction that aims at identifying agreement solution options through an iterative process of making proposals (offers). In Group-Choice Decision Making (GCDM) process, the attributes of these proposals depend heavily on the preference models of the concern agents, and on the knowledge that the agents have about the preference models of their negotiation opponents. Consequently, apposite negotiation models should be able to assist agents to collect preference information of their negotiation opponents, and to integrate this information with their own preferences models in order to identify and make proposals that are most likely to be accepted as the agreement solution options.

Although many negotiation models have been reported in literature, they all fall under two distinctive categories, namely: analytic based models [2, 7] and knowledge based models [1, 3]. In the context of Group-Choice problems in Large-Scale Multi-Agent Systems (LSMAS), negotiation models in literature have the following shortfalls:

- Both categories of negotiation models are developed with an implicitly assumption that agents are always available during the entire negotiation period. This is not realistic in large-scale distributed multi-agents systems, because in such systems agents are terminated or crash without warning.
- Most analytic based agent negotiation models employ techniques that are naturally centralized; this is against the principle of decentralization, which is fundamental to the concept of Multi-Agent Systems (MAS). Moreover, analytic based models require the central processor to have complete information about the preferences of all the negotiating agents. This is impractical for LSMAS.
- Most knowledge based negotiation models result in random behavior (luck of mechanism to track the negotiation process) of the negotiating agents. This behavior results in unnecessary deadlocks in LSMAS. The few knowledge based negotiation models that track negotiation processes are invariability feasible for negotiations involving two agents, such as in the buyer seller negotiation problem.

This paper presents a negotiation model for solving group-choice problems in LSMAS. The model extends the capabilities of the agent negotiation model associated with the **A**gent **N**egotiation **E**ngine for **Co**llaborative **De**cision Making (ANE-CODEM) (see Wanyama and Far [10]). It is based on categorizing negotiation opponents of agents according to the similarity of their preferences. Since the agents focus on making proposal that are acceptable to classes of opponents, instead of dealing with each of the opponents individually, the model presented in this paper enables the agents to address issues associated with many negotiation opponents. This makes the model to be practical for both small and large-scale MAS. Furthermore, the model allows the agents to seamlessly join or leave the negotiation process, which addresses the issue of agents crashing, or being started and terminated without warning.

This paper is arranged as follows: Section 2 presents the related work, and Section 3 described how our negotiation model for LSMAS evolved. Section 4 presents a simulation example that illustrates the capabilities of our negotiation model for LSMAS. Finally, conclusions are given in Section 5.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 463–468.

II. RELATED WORK

Negotiation is a very extensive subject that spans from pre-negotiation to post-negotiation analysis, both at the local and social level. Consequently, considerable amount of work on negotiation is available in literature from different domains, such as operational research, economics, and decision theory [4, 5]. In this section, we present the work that is directly related to our negotiation model.

The analytic based agent negotiation models utilize analytic techniques such as *Game Theory* to determine the solutions that maximizes the social welfare of the negotiating agents [7]. These models minimize communication among the negotiating agents; however, besides the drawbacks associated with LSMAS presented in Section 1, these models invariably have the following general shortfalls:

- The agents have no control over the tradeoffs made during the negotiation process.
- The analytic based models do not follow the natural process of negotiation, where in between offers and counter offers, multiple negation decision variables are traded-off against one another, in order to identify the solution that maximizes the social welfare.

Kraus [8] presents a knowledge based agent negotiation model that implicitly depends on tradeoffs made by the negotiating agents to determine the agreement solution. In the model, the agents evaluate the solution options individually, and then start the process of making offers and counter offers. In between each negotiation round, the agents make tradeoffs aimed at identifying a solution option that is acceptable to all negotiating agents. This model has the following major shortfalls:

- It does not give any guarantees that the agreement solution maximizes the social welfare of the negotiating agents.
- It does not support learning from the offers made by the agent negotiation opponents in order to enable the agents to make offers that are more socially acceptable, as the negotiation progresses; resulting in a random behavior of the agents.
- The agents have no way of knowing whether the negotiation is converging or not.

To circumvent the shortfalls of the analytic models, as well as the shortfalls of the Kraus [8] model, Faratin et al [6] have proposed an agent negotiation model, which depends on utility, similar to the analytic models. Moreover, the model enables the agents to tradeoff during negotiation, like the knowledge-based models. The negotiating agents can utilize the model proposed by Faratin et al even if they have partial information about the solution, thus the model has the potential of enabling the agents to search a larger solution space. However, in the context of LSMAS, the model has the shortfall of being viable for only two negotiating agents such as in buyer-seller negotiation problems. Therefore, the approach of Faratin et al may not be applicable to LSMAS in its current form.

Ray and Triantaphyllou [9] propose a negotiation model that is based on the possible number of agreements and conflicts on the relative importance of the decision variables. However, having different preference functions does not necessarily mean preferring different solution options. Therefore, this model is too inefficient to be utilized in LSMAS. The other shortfalls of this model are the assumptions that the clients of the agents have the same concerns, hence the same set of decision variables, and that the preference models of negotiating agents is public information. In practice, agent clients normally have different concerns, which lead to having different sets of decision variables, as well as preference value functions, and this information is usually private.

This paper presents an agent negotiation model that extends the capabilities of the model associated with ANE-CODEM, in respect of Group-Choice decision making in LSMAS. We call the model the **U**niversal **A**gent **NE**gotiation **M**odel (UANEM), because it is applicable to both small and large-scale MAS. Moreover, the model can be used in a variety of negotiation problems such as Group-Choice negotiation, Seller-Buyer negotiation, and Auction problems. It should be noted that this paper focuses on the use of the model in the Group-Choice negotiation problems. Finally, it should be noted that UANEM is similar to the negotiation model of Faratin et al [6]; except, UANEM utilizes a Game Theory component of ANE-CODEM to support negotiation among *n*-agents.

III. THE BACKGROUND TO THE DEVELOPMENT OF THE UANEM

The UANEM was developed as a result of an evolutionary process that began with the development of a Group-Choice Negotiation Model for Multi-Agent Systems (GCNM-MAS) that is presented in Wanyama and Far [11], and is shown in Figure 1. The capabilities of the model were extended, resulting in a new and more powerful agent negotiation model that is based on the ANE-CODEM (see Wanyama and Far [10]).

A. The GCNM-MAS

We developed the GCNM-MAS for use in a Decision Support System (DSS) for the selection of Commercial-Off-The-Shelf (COTS) products, which we were working on [11]. The main objective of that project was to develop a DSS, which allows both the group and the individual stakeholder processes to be carried out concurrently. Therefore, our main concern was the provision of appropriate user agents for the various stakeholders of the COTS selection process, and the integration of the user information to automatically identify the 'best-fit' COTS products. The automatic negotiation was not met to replace the human decision makers, but to assist the stakeholders to carryout simulation based analysis and ask the 'what if' questions, both at the individual and group levels. At that time we did not mind whether the resulting MAS was centralized or decentralized. Moreover, the COTS selection problem normally involves few (3-10) stakeholders, thus our agent negotiation model did not have to satisfy the requirements imposed by LSMAS.

Figure 1 shows the main features of GCNM-MAS, and the model works as follows:

1. Each user agent j determines the score $\pi_j(i)$ of every solution option i, and sends these scores to the arbitrator agent, which determines the optimal solution option for the negotiating agents using a *Game Theory* model.
2. The Arbitrator agent ranks the solution options based on their *Social Fitness Factors* (G_f), which is a measure of their closeness to the optimal solution option.

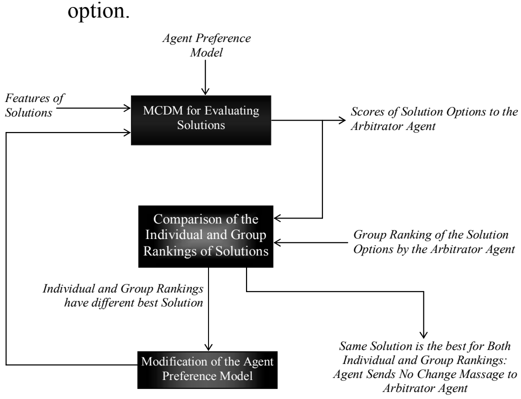


Fig. 1: The GCNM - MAS

3. If the 'best' *Social Fitness Factor* corresponds to the most preferred solution option for all agents, the negotiation ends. However if any of the agents prefers another option, it adjusts its preference model in such away as to improve the score (payoff) of the option with the best G_f. After adjusting the preferences, the agent evaluates all solution options using the new preference model and then sends the new scores of the solution options to the arbitrator agent. This amounts to calling for another around of negotiation.

The above three steps continue until all agents prefer the alternative with the 'best' G_f, or all agent acknowledge that there is nothing they can change to improve their negotiated payoffs without depreciating the G_f of the best fit alternative considerably.

The negotiation model in Figure 1 turned out to be very unreliable. Whenever the arbitrator agent was unavailable, it would not be possible to carry out any group processes. This was very frustrating since we had designed our negotiation model in such a way as to support asynchronous decision making, where agents that are not available at some stage of the negotiation process can catch up with the others at a later stage without being at an advantage or a disadvantage. Moreover, the model assumes environments where only the grand coalition maximizes the utility of the agents. Yet, in practice forming a grand coalition does not guarantee maximum utility for the involved agents. Finally, the negotiation model in Figure 1 does not follow the natural process of negotiation, where agents trade offers and counter offers. Instead the model relies on arbitrator to resolve the differences between the agents.

B. Agent Negotiation Model Based on ANE-CODEM

To address the shortfalls of GCNM-MAS, we modified the agent negotiation engine, and came up with ANE-CODEM [10]. In the negotiation model based on ANE-CODEM, the acceptance of offers is based on a factor called the Acceptance Factor, which is calculated from the amount of tradeoff associated with a solution option (determined using a Qualitative Reasoning mode presented in Wanyama and Far [12]) and the degree of fitness of an offer with respect to the group preferences (determined using a Game Theory mode Presented in Wanyama and far [11]).

Figure 2 illustrates how the *Acceptance Factors* of the solution options are updated. It should be noted that the figure depicts only a single negotiation round. Moreover, Figure 2 shows that if an agreement is not reached by the end of a negotiation round, the final *Acceptance Factors* of the solution options are used in the negotiation engine to modify the preference model of the concern agent in preparation for the next negotiation round. The agent modifies its preference model by adjusting the preference values of some decision variable in such a way as to increase the score of the solution option with the 'best' *Acceptance Factor*; if that solution is not the agent's most preferred, then the modified preference model is used to evaluate the solution option at the beginning of the next negotiation round.

When we employed the negotiation model based on ANE-CODEM in solving Group-Choice problems that involved many (more than 15) stakeholders, the model proved to be inefficient. For example, an agent running on a Personal Computer (PC) with the following specifications: AMD Duron (tm) Processor, 1.10 GHz, 256 MB of RAM), would cause the PC to freeze for up to 5 seconds whenever the agent received the last offer in a negotiation round involving 20 negotiation opponents. Since we designed our agents to run on general purpose PCs and/or servers, this level of resource utilization was unacceptable, because it interfered with other processes running on these machines.

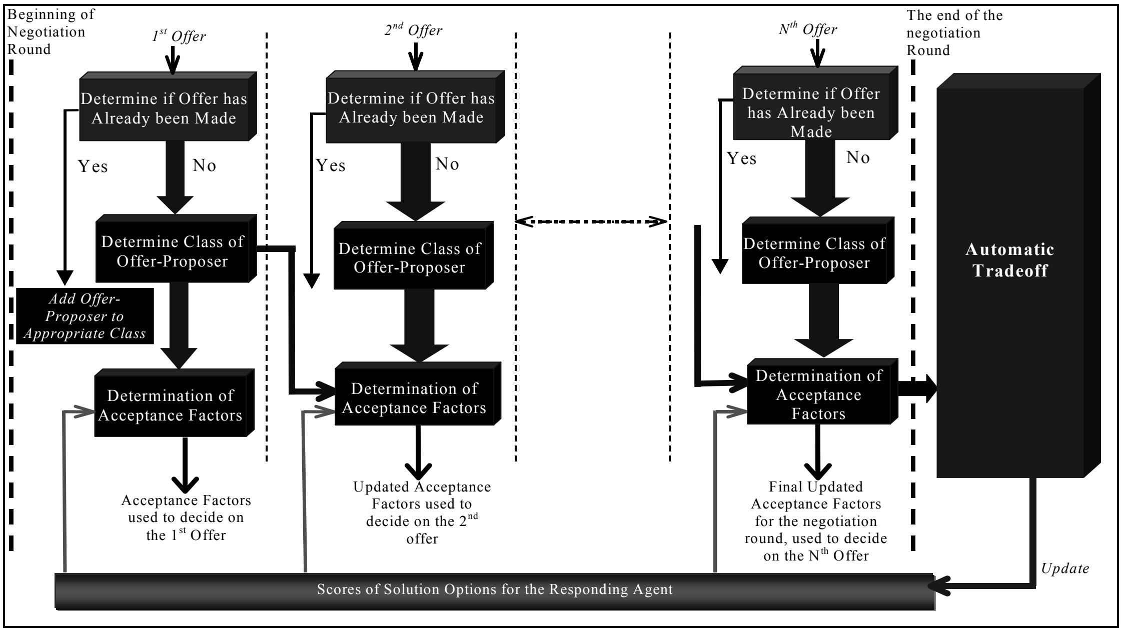


Fig. 2:Agent Negotiation Model Associated with ANE-CODEM

Moreover, such time delays would definitely affect the applicability of the negotiation model to time-constrained Group-Choice problems such as resource management in wireless networks. Therefore, we modified ANE-CODEM to reduce on the computational resource, as well as the time required by agents to respond to offers. The negotiation model that resulted is applicable to both small scale and large-scale MAS, and it can be modified to become applicable to other negotiation problems such as buyer-seller negotiation and auction problems. We therefore refer to this model as the **U**niversal **A**gent **NE**gotiation **M**odel (UANEM).

3.3 The UANEM

To make our agent negotiation model applicable to LSMAS, we reduced amount of processing offers, by enabling the agents to classify their negotiation opponents according to the similarity of their preference models. This was achieved by adding capability to the *Qualitative Reasoning* algorithm of ANE-CODEM to compare offers, as well as the estimated preference models of the negotiation opponents of agents. The resulting agent negotiation model that we refer to as AUNEM is similar to the model shown in Figure 2, but instead of the input to the *Game Theory* model being the estimated scores of the solution options with respect to all the negotiation opponents of the concern agent, as well as the actual scores of the solution options for the concern agent; it is a set of the scores of the solution options associated with the various classes of the negotiating agents, and the number of agents in each class. This compresses the input data to the *Game Theory* model, resulting in a reduction of the computational resources and time required by the agents to respond to offers.

The UANEM can be viewed as a version of the model in Figure 2 that has memory of previous offers, and that has the ability to classify the negotiation opponents of agents according to the similarities of their offers. On receiving an offer, agents in a negotiation process that is based of UANEM are required to check the offer to determine if the same offer has been previously received in the current negotiation round. This results in two scenarios:

- The offer has previously been received: in this case the agent proposing the offer is added to the classes of agents that is associated with its offer, and the number of agents in each class, as well as scores of the solution options that corresponding to every agent class are sent to the Social Welfare Component of the negotiation engine of the concern agent.
- The offer has not previously been received: in this case, the preference model of the proposing agent is estimated, then it is compared with the representative preference models of the existing agent classes. If it is found to be similar to one or more of the category representative preference model(s), the agent is marked as a member of the class whose preference model is most similar to the preference model of the agent. However, if the preference model of the proposing agent is not

similar to any of the representative preference models of existing agent categories, the proposing agent is marked as the first member of a new agent class, and its preference model is labeled the representative preference model of the new agent class.

It should be noted that the level of similarity (ω) between two preference models could be set anywhere between the extremes of 0% and 100%. Where the 100% setting means that for two preference models to be similar, they must have the same decision variables and the same preference value functions. In other words, the two preference models must be identical. On the other hand, the 0% setting of implies that the preference models being compared do not have to have anything in common to be treated as being similar. In fact, with a 0% setting there is no need to go through the process of memorizing previous offers or comparing preference model. The 0% setting reduces the UANEM to the model proposed by Kraus [8]. In that model, agents do not process the offers of their opponents, and adjust their preference models randomly at the end of every negotiation round.

IV. SIMULATION EXPERIMENTS

In these experiments, agents were required to select a Commercial-Off-The-Shelf (COTS) product to be used in the development of a web-shop, from a set of eight solution options. The agents evaluated the solution options based on the preference models made up of twelve predefined decision variables, and the initial preference value functions of the agents were generated using a truncated random number generator.

Three types of agent negotiation models were tested in the experiments: the model proposed in Kraus [8], the agent negotiation model associated with ANE-CODEM (see Section 3.2), and the UANEM. In all experiments, we kept the number of solution options constant (eight solution options), and the number of negotiating agent was increase from 2 to 50 in steps of 1. For each number of agents, we ran the simulation one hundred times, noting the negotiation rounds, and the time taken by one of the two agents with which the simulation started (Agent a), to process the last offer in every negotiation round. The last offers in the rounds are targeted because they involve processing the preferences information of all the negotiating agents; thus resulting in maximum offer processing time. For the UANEM, we carried out simulations with the value of ω set to 0%, 50% and 100%. Moreover, for simplicity we made the following assumptions with regard to the second version of our agent negotiation model: All negotiating agents subscribe to the grand coalition, and every agent is totally committed to maximizing the utility of the grand coalition.

The simulation measurements were carried out on a computer that has the following specifications: AMD Duron (tm) Processor, 1.10 GHz, and 256 MB of RAM. The MAS that we tested in the simulations was developed using Java, and it ran on windows XP machines with Java Run-time Environment (JRE 1.4.2).

A. Results

Figure 3 shows the variation of the maximum number of negotiation rounds with the number of agents involved in the negotiation process.

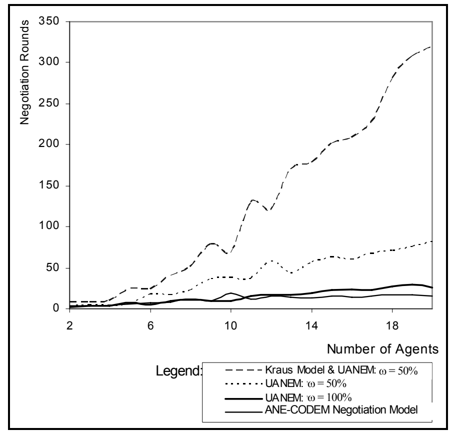


Fig. 3: Variation of Negotiation Rounds with the Number of Negotiating agents

Figure 4 presents the variation of the maximum offer processing time, with the number of agents involved in the negotiation process, for the negotiation model associated with ANE-CODEM and for UANEM with ω set to 100%.

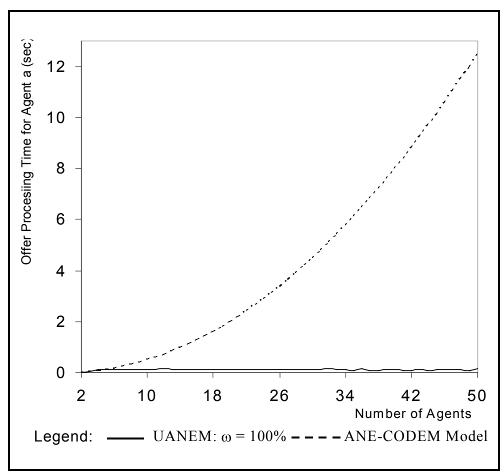


Fig. 4: Variation of the Offer Processing Time with the Number of Negotiating Agents for UANEM with $\omega = 100\%$ and the ANE-CODEM Model

Note that each of the maximum offer processing time in the figure is an average of the time for processing the last offers of every negotiation round in one hundred negotiation simulations.

B. Discussion of Results

Figure 3 reveals that the negotiation model proposed in Kraus [8] is synonymous with the UANEM with the similarity level (ω) set to 0%. Moreover, the Figure shows that the performance (in term of negotiation rounds) of the UANEM with ω set to 100% is comparable to that of the negotiation model associated with ANE-CODEM. Any other setting of ω results in a negotiation-rounds performance that lies between that of the UANEM with $\omega = 0\%$ (Kraus's Model) and that of the UANEM with $\omega = 100\%$ (see performance of the UANEM with $\omega = 50\%$). For Kraus's negotiation model [8], the number of negotiation rounds increases sharply with increasing the number of agents involved in the negotiation (see Figure 3). This makes it inappropriate for LSMAS.

Kraus's agent negotiation model [8] does not require the agents to carryout any processing of the offers that they receive. This saves processing time, but it results in random behavior of the agents, leading to poor or no control of the dynamics of the negotiation process. On the other hand, the negotiation model associated with ANE-CODEM requires agents to process offers that they receive in order to identify appropriate counter offers. This controls the dynamics of the negotiation process. However, processing of offers results in offer processing time that increases sharply with increasing the number of agents involved in the negotiation process (see Figure 4). This makes this model inappropriate for LSMAS. Furthermore, Figure 4 shows that the UANEM (ω set to 100%) results in offer processing time that does not vary significantly with the number of agents involved in the negotiation process, implying that the mode is applicable to LSMAS.

V. CONCLUSION AND FUTURE WORK

This paper presents an agent negotiation model for GCDM in LSMAS. Moreover, the paper describes how the negotiation model for LSMAS was derived from a simple centralized negotiation model. The simulation results presented in this paper show that the negotiation model proposed in Kraus [8] and the negotiation model associated with ANE-CODEM represent the two extremes of knowledge based negotiation models. That is, in the context of LSMAS, Kraus's model is associated with minimum (zero) offer processing time and maximum number of negotiation rounds. On the other hand, the negotiation model associated of ANE-CODEM is associated with maximum offer processing time and minimum number of negotiation round. Furthermore, the simulations reveal that the UANEM is associated with low offer processing time (close to Kraus's model) and low negotiation rounds (close to the negotiation model of ANE-CODEM), making it suitable for LSMAS.

From Figures 3 and 4, it is noticed that the offer processing time and the number of negotiation rounds vary in opposite directions with the variation of the similarity level (ω). Therefore, in the future we would like to establish the optimal similarity levels associated with different agent negotiation situations.

REFERENCES

[1] H. Raiffa, The Art and science of Negotiation. Cambridge: Harvard University Press, USA, 1982.

[2] M. Barbuceanu and W. Lo, A Multi-attribute Utility Theoretic Negotiation Architecture for Electronic Commerce. In the Proceedings of the Fourth International Conference on Autonomous Agents, Barcelona, Spain, 2000. (Agents 2000), pp. 239-247.

[3] M. Yokoo and K. Hirayama, Distributed Constraint Satisfaction Algorithm for Complex Local Problems. In the Proceedings of the Third International Conference on Multi-Agent Systems. IEEE Computer Society Press, 1998, pp. 372-379.

[4] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, Automated Negotiation: Prospects Methods, and Challenges, International Journal of Group Decision and Negotiation 10(2), pp. 199-215, 2001.

[5] N. R. Jennings, P. Faratin, M.J. Johnson, P. O'Brien, and M.E. Wiegand, Using Intelligent Agents to Manage Business Processes. In the Proceedings of the Practical Application of Intelligent Agents and Multi-Agent Technology Conference, London The United Kingdom, 1996. PAAM 1996, pp. 345-360.

[6] P. Faratin, C. Sierra, N.R. Jennings Using similarity criteria to make issue trade-offs in automated negotiations. Artificial Intelligence, 2002, pp 142, 205-237.

[7] S Kraus 1997. Negotiation and Cooperation in Multi-agent Environment. Artificial Intelligence 94 (1-2), pp. 79-97.

[8] S. Kraus, Strategic Negotiation in Multiagent Environments. Cambridge: Massachusetts Institute of Technology Press, 2001, 17-29.

[9] T. G. Ray, E. Triantaphyllou, "Theory and methodology:- Evaluation of Rankings with regard to the Possible number of Agreements and Conflicts", *European Journal of Operational Research*, pp. 129-136, 1998.

[10] T. Wanyama and B. H. Far, An agent Negotiation Engine for Collaborative Decision Making, *The Modeling Decisions for artificial Intelligence Conference*, Tarragona, Spain, April 2006, LNAI 3885, pp. 46-57.

[11] T. Wanyama and B. H. Far, Multi-Agent System for Group-Choice Negotiation and Decision Support. In *the proceedings of the 3rd Workshop on Agent Oriented Information Systems,* New York USA, July 2004. AAMAS 2004, pp. 194-208.

[12] T. Wanyama and B. H. Far, Qualitative Reasoning Model for Trade-off Analysis, *The Modeling Decisions for artificial Intelligence Conference*, Tsukuba Japan, July 2005, LNAI 3558, pp. 99-109.

Sensitivity analysis of parallel applications to local and non-local interference

Vaddadi P. Chandu[*], Karandeep Singh[#]
[*]Georgia Institute of Technology, Atlanta GA
vchandu@cc.gatech.edu
[#] University of Maryland, College Park MD
ksingh@mail.umd.edu

Abstract

The environment in which a parallel application is executed has high impact on the performance of the application due to interference caused by various factors in the execution environment. A detailed understanding of the sensitivity of the application to the parameters describing the execution environment can be of great help in (a) predicting a suitable target machine model for the application, (b) predicting its performance on the target machine, and (c) any algorithmic bottlenecks. In this paper, we analyze a suite of parallel applications for their sensitivity to local and non-local interference arising due to various factors in a parallel environment. We create a test bed consisting of five different parallel applications taken from different sources and analyze their sensitivity to single node and multi-node perturbations and show that parallel applications can behave very differently under different conditions of interference in the environment in which they are running. The main contributions of this paper are: (a) studying a suite of parallel algorithms for their sensitivity to local and non-local interference, (b) demonstrate that an application can behave differently to different interference levels in the environment, (c) demonstrate that the sensitivity of an application can be quantified as its absorption ratio at a given interference level.

1. Introduction

The performance of a parallel application in distributed memory machines is highly dependent on the local and non-local interference existing in the environment which can be due to operating system interference or communication latency. Interference of any kind in the environment is highly detrimental to the performance of a parallel application. Hence, the ability to understand the sensitivity of a parallel application to interference at different levels is very important to understand its behavior in that environment. One of the techniques to analyze the sensitivity of a parallel application is to simulate it for compute time and messaging latency and see the effect on the execution time by changing these parameters. Petrini et al in [8], describe a method to analyze the performance of parallel applications on a parallel machine through modeling the application and study its behavior before actually building the machine. In [11], a methodology to analyze a parallel application through trace-based analysis is presented. In this method, the message-passing events in a parallel application are extracted during its execution and are linked in the same order as they occurred during the execution. This method provides an actual execution model of the application which is free from any assumptions or conceptual errors. Dimemas [3] and Vampir [14] are also similar methodologies but this approach significantly differs from them in the way that interference from local and non-local sources can also be simulated.

Trace-based analysis of a parallel application can be visualized as creating a message-passing graph (although it need not be actually created) in which any two message-passing events are interconnected with an edge that represents communication in the original application during its execution time. Each edge contains a start time stamp, an end time stamp, and an edge weight. Changing the weights on the edges simulates perturbations corresponding to local or non-local interference such as operating system noise or communication latency. Since the trace is extracted from the actual execution of the application, this method guarantees the correctness of the application and retains the actual ordering of the messages in the application.

In this paper, we focus on analyzing a suite of parallel applications for interference from local and non-local sources such as CPU throttling, network congestion, overloading of a single node, and misconfiguration of the operating system on all nodes. Although, our work is in the context of MPI [4] library, this method can be easily extended to any implementation ARMCI [6], PVM [12] of the message passing interface. The rest of the paper is organized as follows. In Section 2, we describe the message-passing graph concept. In Section 3, we describe our experimental dataset, and in Section 4 we present our results. In Section 5, we present the conclusions and future research.

2. Message Passing Graph Concept

The execution behavior of a parallel application, which is written using a message passing library, can be modeled as a message-passing graph by tracing it during runtime. The fundamental idea behind modeling it as a message passing graph is that, during execution, most parallel programs alternate between computation and communication.

Hence, by time stamping the events of computation and communication and joining them through directed edges, an exact representation of the execution model of the

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 469–474.

application can be produced. In such a message-passing graph, the edges connecting any two events in the same trace are called as local edges and the edges connecting any two events in different traces are called as message edges.

After representing the execution of an application as a message-passing graph, perturbations can be injected into a local or a message edge and propagated along the graph ensuring that the execution order of all events is preserved and no two events violate each other's time stamps. However, if there room to absorb the perturbation instead of propagating it without violating the execution order, it is absorbed. Perturbing a message-passing graph in this manner increases the execution time by a definite amount thereby providing an insight into the amount of interference that the application can absorb. In this manner, the sensitivity of the application to that perturbation can be computed.

An application can be traced during execution time by wrapping the MPI primitives with a lightweight PMPI wrapper using the standard PMPI interface which is defined in the MPI specification. Each node generates an event trace that includes all the events that occurred at that node, a time stamp for each event, the type of the event, and corresponding metadata. It is important to mention here that although events are paired in a message-passing graph, they only represent the ordering of the execution and not the actual synchronized distributed-clock timing. However, this is not an issue in our analysis because we require only the actual execution ordering of the events and not the actual synchronized timing. Another vital issue is to confirm the correctness of the concept. Since the injected perturbations follow the same path as in the original execution and propagate through the message passing graph along the edges, the dependencies are automatically taken care of and hence the correctness is preserved.

However, it is important to note that there are two classes of applications where this method may not provide accurate results. First, the class of applications that is non-deterministic, such as the Branch and Bound algorithm where the communication pattern is highly non-deterministic, and second, the class of applications which is embarrassingly parallel such as SETI@home [7], where there is very little or practically no communication between the nodes. In both cases tracing the application will not provide enough information to generate the correct execution model of the application and hence cannot provide accurate results. However, this method works well for all the applications that do not fall in the above two classes.

2.1 Analyzing the sensitivity of an application using trace-based analysis method

After creating the execution model of an application, the sensitivity of the application to local and non-local interference on particular platform can be measured by using the signature of that particular parallel platform. Signature of a parallel platform can be constructed using microbenchmarks. Microbenchmarks are capable of probing a given platform to a very minute detail. By assuming a distribution on these results, a sufficiently accurate parameter set for a given parallel platform can be estimated. One such microbenchmark to estimate the operating system noise is the Fixed Time Quantum microbenchmark [10], which probes the operating system for periodic perturbations in a large number of fine grained workloads. Using a simple message-passing program as suggested in [5] is another method to estimate the operating system noise for a particular platform.

Communication parameters can be benchmarked in a similar way. Intel Cluster Toolkit [13] provides a suite of benchmarks which provides three classes of benchmarking functionalities- Single Transfer, Parallel Transfer, and Collective. The single transfer class benchmarks the performance of a single message transfer between two processes using PingPing and PingPong style messaging. The parallel transfer class benchmarks the performance the message transfer performance under global load using a periodic chain and periodic exchange pattern styled messaging and the collective class benchmarks the performance the performance of the collective primitives of the message passing interface using the style as defined by the collective primitives in the MPI library.

3. Experimental Test Bed

In our experimental test bed, we consider five different applications for the sensitivity analysis using the trace based analysis method. The applications are Parallel Sorting using Regular Sampling [9], Cycle Detection of Partitioned Planar Digraphs [1], Parallel Selection and Median Finding [2], Parallel Heap Sorting, and A Grid based problem-Jacobi iteration, each of which is described in the next subsection. All the applications in our test bed are written using the MPI library (version 1.2.6) and are compiled using level three optimization enabled for a distributed memory machine which uses Infiniband interconnect in Linux environment. Each node is a 3.20 GHz Intel Xeon dual core CPU with 1024 KB cache memory and 1 GB main memory.

3.1 Parallel Sorting using Regular Sampling

Parallel Sorting by Regular Sampling is a sorting algorithm for distributed memory machines. This algorithm uses techniques to reduce the bus and memory contention by ensuring a good pivot selection through regular sampling of the data. It fits into the partition-based sorting methodology, where the data set is partitioned into smaller subsets such that all elements in one subset are no greater than any element in another, and each subset is sorted in parallel.

This algorithm, through regular sampling of the data, is capable of finding pivots to partition a dataset into similar sized smaller datasets. Due to uniform partition of the data, load balancing is also uniform. This algorithm consists of the following three phases.

Consider p processors. In the first phase, the input dataset is uniformly distributed among the p processors using sequential Quicksort algorithm. Each one of the p processors sorts the list of data elements assigned to it. From each of these lists (which are p in number), p-1 samples are chosen such that they are evenly spaced in the list. Each one of the (p-

1) samples is sorted locally using the sequential Quicksort algorithm. In the second phase, from the p(p-1) lists, evenly spaced p-1 pivots are chosen, and all processors synchronize with the ordered sublists. In the third and the last phase, each processor performs a p-way merging of the i^{th} sorted sublists of the p lists. Since the contiguous blocks are now located in different regions, each processor can work independent of the other and hence this algorithm achieves good parallelism.

3.2 Cycle Detection of Partitioned Planar Digraphs

This is a parallel method to detect the cycles in planar digraphs. Cycle detection is applied in the areas of computational physics, mechanics, and fluid flow. The algorithm underlying this application consists of the following three phases.

In the first phase, which is called as the *discovery* phase, each vertex in the graph tries to discover the set of trans-arcs (arcs which span across two processors) that have a terminal vertex assigned to it. All incident arcs and vertices that are found to be acyclic are removed from the graph in this phase. Upon finding a cycle, the algorithm halts.

In the second phase, which is called as the *express* phase, an exit or entrance vertex is created for all the terminal or initial vertices of the trans-arcs and is assigned to the same processor. The newly formed graph is called as the Express graph. The trans-arcs are moved to entrance or exit vertices in the express graph and for all entrance vertices an express arc is added to the graph.

In the third and final phase, which is called as the *merge* phase, the graphs created in the second phase are iteratively combined until only one graph remains on one of the processors. This phase involves aggressive pruning of the vertices and arcs that are acyclic.

3.3 Parallel Section and Median Finding

This application offers a solution to the problem of finding the median for a given set of elements distributed across a parallel machine. In this method, first, the input is distributed across p processors with an irregular distribution which is not known a priori. In order to redistribute the data across the processors, all elements are ranked in order along all processors. After the redistribution process, the selection process begins. The selection process makes no assumptions about either the number of elements that are held by each processor or about the distribution of the values on any processor. Through recursion, a "pivot" element is chosen form the collection so that the input is partitioned into two, where one partition contains all elements less than or equal to the pivot and the other one contains all elements that are higher than the pivot.

For example, out of n-elements in the lower partition, if the pivot index is *i* and is less than or equal to *n*, then the recursion is performed on the lower partition, else on the other partition. The pivot element is chosen by choosing a local median at all processors and then finding a median of all local medians. In order to balance the work among processors, a dynamic redistribution technique is employed on the dataset.

3.4 Grid based application: Jacobi iteration

The Jacobi iteration computes 2-D Laplace's equation. The problem can be formulated as follows. Given a grid describing the decomposition of a mathematical domain (possibly on a physical object) with an initial state, solve a PDE or other equation set on it and determine the result such that the error threshold is within a given permissible limit ε. This problem can be solved using p-processors in the following manner. Partition the initial grid dynamically into p-blocks. Let each processor compute the Tij value at each grid point in its block according to the pre-determined equation. To compute the boundary values of its block for which it requires the boundary values of its neighboring processors, let a processor communicate with the neighboring processors to receive the values.

Compute the cumulative error Ξ based on some pre-determined function from the value of the grid in the current and previous iterations. Continue until $\Xi < \varepsilon$. We consider it as a representative of the class of grid-based applications because many applications in physics such as lattice percolation and heat dissipation in a given layout follow closely the grid based approach and the iterative methodology of this problem.

3.5 Parallel Heap Sorting

In this application, we implemented a parallel heap sort. We parallelize the problem by dividing the input work set evenly among the available number of processors (say p). For simplicity, the input set is rounded off to the nearest value that is divisible by p. Each processor sorts its local array using a sequential heap sorting algorithm. Locally sorted arrays are merged together, in a log-tree fashion. This process continues until all the process converges at the root of the tree when all the elements are sorted.

4 Experimental results and analysis of the results

The input to the applications and the number of processors used for each one of them is as follows. (a) Parallel Sorting using Regular Sampling: 2^{19} integers on 16 processors, (b) Cycle Detection of Partitioned Planar Digraph: 2^{14} vertices on 16 processors, (c) Parallel Selection and Median Finding: 2^{20} integers on 8 processors, (d) Jacobi Iteration: 2 million grid points on 16 processors with an error threshold ε of 0.1, and (e) Parallel Heap Sorting: 50 million integers on 16 processors.

For our analysis, we use Chama which implements the concepts that we described in Sec. 2. We inject a perturbation of 10 s into the applications in our test bed and analyze their sensitivity considering the following two cases. First, we perturb all nodes with equal probability ranging from 0.01 to 1.0 with increments of 0.01. Second, we perturb one node with a non-zero probability again ranging form 0.01 to 1.0 with increments of 0.01 and all other nodes with a constant zero probability. The node that is perturbed with non-zero probability becomes the slow node in the system. At each probability interval, we output the mean and total delay due to injected perturbations. From these values, we infer the sensitivity of the application by computing the absorption ratio, which we define as the ratio of amount of delay absorbed (mean delay subtracted from the total delay) to the total delay. An application with a high absorption ratio implies that the application can tolerate interference without significant performance degradation and hence insensitive to the interference and an application with a low absorption ratio implies that it cannot tolerate interference without performance degradation and hence is very sensitive to the interference.

Figures 1 to 5 show our analysis of the applications in our test bed. Fig. 1(a) shows the sensitivity of the application *Parallel Sorting using Regular Sampling* when all nodes are equally perturbed and Fig. 1(b) shows its sensitivity when a single-node is perturbed with non-zero probability and all others are perturbed with zero probability. In the single-node case, the sensitivity decreases linearly with increasing perturbation probability, and in the all-node case, the sensitivity decreases in a non-linear manner in the beginning, reaches a minimum value at a perturbation probability of 0.4, and again starts rising. This behavior indicates that the application is highly sensitive to any interference (single-node or all-node) in the execution environment and hence may perform very poorly in a noisy environment.

On the other hand, Fig. 2(a), which represents the application, *Cycle Detection of Partitioned Planar Digraphs*, indicates that the application is very sensitive to interference when all nodes are equally perturbed. From this, we can infer that if there is congestion in the communication network, or poorly configured operating system on all nodes, the performance will significantly degrade. However, in Fig. 2(b), a low sensitivity to interference on one node indicates that the performance of this application will not significantly degrade due to interference on a single node. Hence, factors such as a misconfigured operating system on a single node may not have a great impact on the performance of this application.

In the application, *Parallel Section and Median Finding*, we see a high sensitivity to interference on all nodes (Fig. 3(a)) and a low sensitivity to interference on a single node (Fig. 3(b)). However, as the probability of interference on the single node is increased, the application gradually becomes more sensitive starting from perturbation probability of 0.4 and finally becomes highly sensitive at a probability of 1.0. This indicates that this application can tolerate noise in a single node without degradation up to a probability of 0.4 beyond which we will see degradation in the performance.

In the grid based application, *Jacobi iteration*, Fig. 4(a), (b) we see a fairly uniform sensitivity to both single node and all-node perturbations for all probabilities. However, the sensitivity to interference in a single node in this case is higher than the sensitivity to interference in all nodes. From this observation, we infer that a single poorly performing node will have more impact on the performance of this application.

In *Parallel Heap Sorting* (Fig. 5), we see that it is extremely sensitive to all-node perturbation for all probabilities, but is slightly less sensitive to perturbation on a single node. This implies that even a small interference in the environment will have a large impact on the performance of the application. However, the application can take some noise on a single node and without much degradation in its performance.

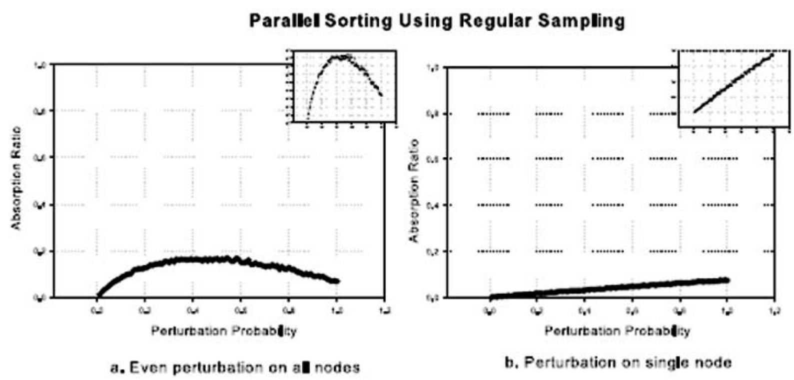


Figure 1. Sensitivity of the application *Parallel Sorting using Regular Sampling***. In (a) the application trace is perturbed on all nodes equally, and in (b) the application trace is perturbed with non-zero probability on one node and with zero probability on all other nodes. A low absorption ratio indicates high sensitivity to interference.**

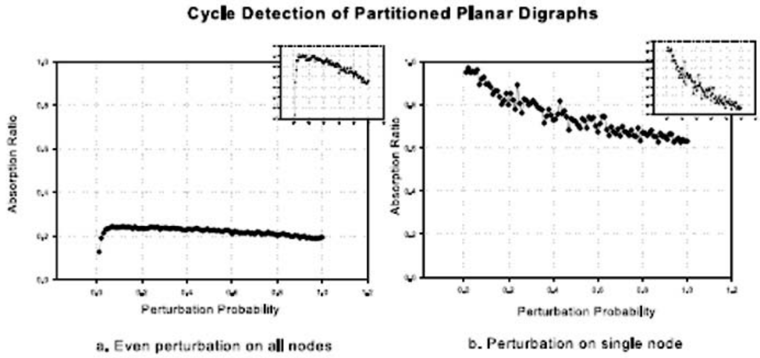


Figure 2. Sensitivity of the application *Cycle Detection of Partitioned Planar Digraphs*. **In (a) the application trace is perturbed on all nodes equally, and in (b) the application trace is perturbed with non-zero probability on one node and with zero probability on all other nodes. This application is less sensitive to single node perturbation but is highly sensitive to all-node interference.**

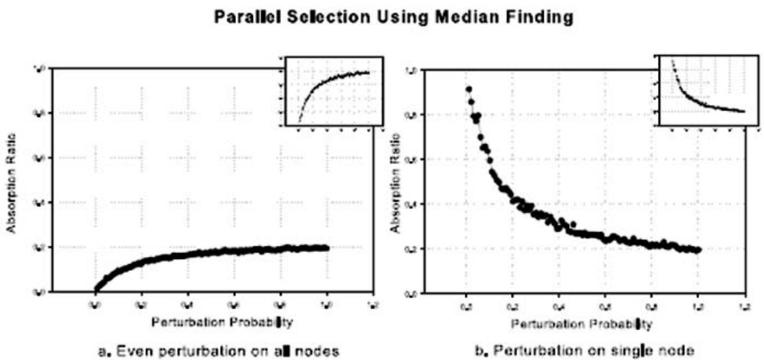


Figure 3. Sensitivity of the application *Parallel Section and Median Finding*. **In (a) the application trace is perturbed on all nodes equally, and in (b) the application trace is perturbed with non-zero probability on one node and with zero probability on all other nodes. This application is highly sensitive to all node interference, but is fairly immune to low interference level on a single node.**

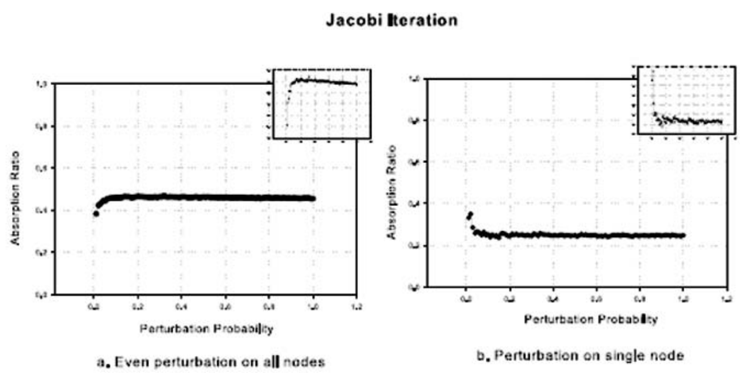


Figure 4. Sensitivity of the application *Grid based method: Jacobi iteration*. **In (a) the application trace is perturbed on all nodes equally, and in (b) the application trace is perturbed with non-zero probability on one node and with zero probability on all other nodes. This application is uniformly sensitive to both single-node and all-node perturbation, however it is more sensitive to single-node interference.**

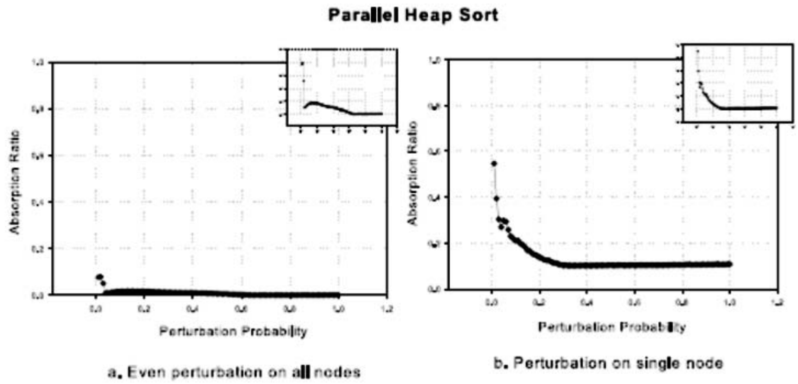


Figure 5. Sensitivity of the application *Parallel Heap Sorting*. **In (a) the application trace is perturbed on all nodes equally, and in (b) the application trace is perturbed with non-zero probability on one node and with zero probability on all other nodes. This application is sensitive to both all-node and single-node interference; however, it is more sensitive to interference on all nodes.**

5 Conclusion and future research

We analyzed a suite of parallel applications for their sensitivity to local and non-local interference. We also showed how to estimate the performance of a parallel application for a particular platform given the signature of the platform. The work can be extended to performance prediction of an application under specific interference conditions on a target machine without actually executing the application on the machine but using the machine signature and injecting the perturbations that represent the target environment. Currently, the simulator we are using to analyze the execution traces is in development phase and supports many but not all of the MPI primitives. Our future work will be to add more functionality to the simulator and develop micro-benchmarks for generating machine parameter sets to be used for our analysis.

References

[1] D. Bader. A practical parallel algorithm for cycle detection in partitioned digraphs. *Technical Report UNM/AHPCC99-013*, 1999.

[2] D. Bader and J. Ja'Ja'. Practical parallel algorithms for dynamic data redistribution, median finding, and selection. *10th International Parallel Processing Symposium (IPPS 96) Conference, Honolulu, HI, April 15-19*, pages 292.301, 1996.

[3] R. M. Badia, J. Labarta, J. Gim´enez, and F. Escal´e. DIMEMAS: Predicting MPI applications behavior in Grid environments. *Workshop on Grid Applications and Programming Tools, 8th Global Grid Forum (GGF8), Seattle, WA*, 2003.

[4] Message Passing Interface Forum. MPI: A message-passing interface standard. Technical Report UT-CS-94-230, University of Tennessee, 1994.

[5] R. Mraz. Reducing the variance of point to point transfers in the IBM 9076 parallel computer. In *Proceedings of the 1994 conference on Supercomputing*, 1994.

[6] J. Nieplocha and B. Carpenter. Armci: A portable remote memory copy library for distributed array libraries and compiler run-time systems. *Lecture Notes in Computer Science*, 1999.

[7] U. of California. http://setiathome.berkeley.edu/.

[8] F. Petrini, D. Kerbyson, and S. Pakin. The Case of the Missing Supercomputer Performance: Achieving Optimal Performance on the 8,192 Processors of ASCI Q. In *Proceedings of Supercomputing 2003*, 2003.

[9] H. Shi and J. Schaeffer. Parallel sorting by regular sampling. *Journal of Parallel and Distributed Computing*, 14(4):361. 372, 1992.

[10] M. Sottile and R. Minnich. Analysis of Microbenchmarks for the Performance Tuning of Clusters. In *Proceedings of Cluster 2004*, 2004.

[11] M. J. Sottile, V. Chandu, and D. A. Bader. Performance analysis of parallel programs via message-passing graph traversal. *20th IEEE Int'l Parallel and Distributed Processing Symp. (IPDPS), Rhodes Island, Greece, April 25-29*, 2006.

[12] V. S. Sunderam. PVM: a framework for parallel distributed computing. *Concurrency, Practice and Experience*, 2(4):315. 340, 1990.

[13] I. web page. www.intel.com/software/products/cluster/clustertoolkit/index.htm.

[14] V. web page. http://www.pallas.com/e/products/index.htm.

Performance Evaluation of a Xen-based Virtual Environment for High Performance Computing Systems

Vaddadi P. Chandu*, Karandeep Singh#
*Georgia Institute of Technology, Atlanta GA
vchandu@cc.gatech.edu
University of Maryland, College Park MD
ksingh@mail.umd.edu

Abstract

Virtualization is becoming an increasingly popular method to achieve software-based solution for sharing hardware infrastructure in a completely isolated manner. Xen virtual machine monitor (an open source software) is becoming a popular resource to implement virtualization for managing multiple operating systems instances within one physical computing node. Current research has focused on migrating these instances between nodes during runtime. This capability is useful for numerous activities, such as fault management, load balancing, and low-level system maintenance. In this paper, we evaluate the performance of Xen Virtual Machine Monitor for high performance computing (HPC) systems and discuss its suitability for 1) easy management of applications (e.g., automatic load balancing of MPI application processes), and 2) easy management of the HPC architecture (e.g., automatic migration of OS instances away from physical resources that have been predicted to fail in the near future).

1. Introduction

Virtualization presents the illusion of many smaller virtual machines, each running a separate operating system instance on the same machine. Such a virtualized environment provides isolation, security, low performance overhead, and supports heterogeneous applications [4]. For large enterprises, where on-demand capabilities are highly desirable, such a virtualization technique is very helpful in building an ideal solution for server and application consolidations. Virtualization allows higher system utilization through better allocation and de-allocation of resources dynamically to the participating applications in a completely isolated manner. However, currently these efforts are not directed at supporting HPC architectures, SSI [8], or parallel programming models that inject dependencies into the migration (e.g., message passing with MPI, global memory accesses in UPC). In this paper, we analyze the issues that confront in using the virtualization infrastructure on high performance architectures. We use Intel Cluster Toolkit [7] for benchmarking purposes and Xen hypervisor system [1, 2, 3] for providing a virtualized environment. The rest of the paper is organized as follows. In Section 2 we briefly describe the Xen hypervisor system and the Intel Cluster Toolkit benchmarking suite. In Section 3 we discuss our experimentation methodology. In Section 4 we discuss our results and we present the conclusions and future research in Section 5.

2.1 The Xen Hypervisor

Xen is a virtual machine monitor which provides a base for running many operating system instances on a single physical machine. Virtualization is known from Mainframes but was unsupported on x86 architectures until Xen was developed. Xen hypervisor acts as a layer between the underlying hardware and various operating systems (also known as guest operating systems) that are running on the hypervisor. The job of the hypervisor is to provide transparent and completely isolated image of the underlying hardware to each one of the guest operating systems. To achieve this goal, Xen does not aim at a complete virtualization of the physical hardware, but instead, uses a technique known as 'paravirtualization' where the guest operating systems are modified accordingly. However, the modifications are restricted to the operating system core and are completely invisible to the user. Xen appears as a kernel modification to the system administrator where the host and guest operating systems both carry kernel patches.

There are four principles governing the design of Xen. First, support for unmodified binaries. Second, support for full operating systems. Third, high performance and strong resource isolation. Fourth, completely hiding the effects of resource virtualization from guest operating systems. These design principles allow Xen to be useful on a wide variety of platforms ranging from large enterprises to research solutions such as computing Grids [9].

Fig. 1 shows the Xen architecture. The usual notation for guest operating system is DomU (short for User Domain) and for hypervisor is Dom0 (Domain0, which indicates full privilege). The Xen hypervisor allows each guest operating system a definite level of authority but ensures that all critical actions go through the hypervisor. For example, paging from guest operating system has direct access to the hardware page tables, but page updates are routed through the hypervisor. Exceptions from a guest operating system are registered through the hypervisor. Segmentation cannot install segment descriptors that are fully privileged. I/O devices are virtualized and asynchronous I/O rings are used to transfer data through the I/O devices which in turn have to go through the hypervisor.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 475–481.

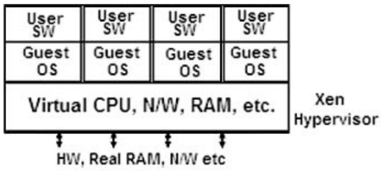


Fig. 1. Architecture of the Xen Virtual Monitor

2.1.1 Xen CPU management

The x86 architecture supports different privilege levels in hardware. It allows four different rings of protection, with highest privilege given to ring0 and lowest to ring3. Xen places the hypervisor in ring0 and the guest operating systems in ring1. Placing the guest operating system at a lower level prevents it from executing privileged instructions without the knowledge of the hypervisor. Instructions such as yielding the processor during idle cycles and installing new page table are all routed through the hypervisor. This guarantees isolation of the privileged instructions. CPU scheduling, which is another important issue in CPU management is handled through Borrowed Virtual Time algorithm [12]. This allows low-latency dispatch through virtual-time warping in which a newly woken domain is favored.

2.1.2 Xen Memory management

Xen allows guest operating systems to allocate and manage the hardware page tables. However, upon creation of each new hardware page table, a guest operating system has to register it with the hypervisor. The hypervisor exists in a 64MB section at the top of every address space which is restricted from the guest operating systems.

Direct writes to memory from a guest operating system are not allowed. All writes are required to be validated by the hypervisor to ensure consistency. Segmentation is also virtualized in a similar way, by validation to the hardware segment descriptor tables.

2.1.3 Xen I/O management

Xen presents a simple abstraction of the hardware devices. It does not perform a complete emulation of the devices. This fulfils the requirements for protection and isolation. Any I/O data is transferred to and from each domain via the hypervisor through asynchronous buffer descriptor rings. The rings are shared between the guest operating systems. Data is sent vertically through the system to allow quick validation checks and achieve high-performance communication. Event delivery to guest operating systems follows a lightweight delivery mechanism where asynchronous notifications are informed by updating a bitmap corresponding to the guest operating system. Guest operating systems can process these callbacks at their discretion.

2.2. The Intel Cluster Toolkit (ICT) benchmark suite

Now we describe the Intel cluster toolkit benchmark suite that we use to evaluate the performance of the Xen hypervisor. The ICT toolkit [7] consists of the following four modules. Intel MPI Benchmarks (IMB), Intel Trace Collector, Intel Trace Analyzer, and Intel MPI Library. For evaluating the performance of the Xen hypervisor for high performance computing systems, we are interested only in the Intel MPI Benchmarks (IMB). ICT uses a profiling methodology similar to the methodology used in [6]. In ICT, a parallel application is linked to a profiling library which adds the required time stamps to the MPI primitives and then calls the actual MPI library implementation. The time stamps are reported back to the user. ICT classifies the MPI primitives in IMB into three classes-Single transfer class, Parallel transfer class, and Collective class.

The benchmarks in the ‘single transfer class’ focus on measuring the performance of a single message transfer between two processes. PingPong and PingPing are two methods IMB uses to measure this. PingPong measures the throughput and startup of a single message between two processes by using MPI_Send and MPI_Recv primitives. The performance is measured as half the total time taken. PingPing also measures the throughput and startup of messages but under non-optimal conditions of oncoming traffic. To achieve this, PingPing uses a combination of MPI_Isend, MPI_Recv by simultaneously issuing the MPI_Isends, and waiting on MPI_Recv. MPI_Waits are introduced for consistency. Due to this difference, for PingPing pure timings are reported, and the throughput is related to a single message. Additionally, the numbers in PingPing, are usually between half and full of the PingPong throughput.

In the ‘parallel transfer class’, the benchmarks focus on measuring message passing efficiency under global load. Each benchmark is repeated with varying message lengths. The timings are measured as an average of the results in the repetitions. A periodic chain pattern [7] through MPI_Send-MPI_Recv, and an exchange pattern [7] in a periodic chain through MPI_Isend-MPI-Recv are used to benchmark the performance. In a periodic chain pattern, each process sends message to its right neighbor and receives message from its left neighbor. Due to its design, for two processes, this benchmark provides the bi-directional bandwidth of the system. On the other hand, in the exchange pattern, each process exchanges messages with both its neighbors, a pattern which occurs while computing the boundary conditions in problems like Parallel Ocean Program [10].

In the ‘collective class’, the benchmarks focus on measuring the performance of the collective primitives of MPI [11], such as MPI_Reduce, MPI_ReduceScatter, MPI_Allreduce, MPI_Allgather, MPI_Alltoall, MPI_Bcast, and MPI_Barrier. In this class, only the raw timing is displayed. The MPI_Reduce and MPI_Allreduce use MPI_SUM as the operator and reduce a vector of float items. In MPI_ReduceScatter, the vector is evenly reduced and split across all the processes. In MPI_Allgather, each process inputs a length of bytes and receives the gathered bytes, which is equal to the length of input bytes multiplied by the number of processes. The MPI_Alltoall functionality is same as the MPI_Allgather with the only difference that the length of the input bytes is equal to the length of the output bytes which in turn is equal to the length of the received bytes as in the case of the MPI_Allgather functionality. The MPI_Bcast functionality simply broadcasts a length of bytes to all processes and MPI_Barrier measures the time for all processes to reach the barrier.

3. Our Experimental Strategy

We create a cluster of computers connected through 100MBPS Ethernet with Xen installed on each one of them. Each guest operating system image is a Debian [13] Linux with MPICH Version 1 installed on it. We choose a 'basic Debian system' with support for very few functionalities, such as the ssh, gcc, and Fortran for the guest operating system image. We chose to install MPICH Version 1 to save some processor cycles by avoiding executing the MPD process manager in MPICH Version 2, although any version of MPICH would work equally well. We use the same guest operating system image for creating any number of virtual domains. All the computers in our test bed consist of a single core 1.7GHz i686 processor (Centrino) with 512MB main memory. A setup similar to our experiment is shown in Fig. 2. πi and λi indicate the internal delays inherent in the Xen operating system and β indicates the communication delay between two Dom0s. In high performance computing systems, parallel applications are executed in the layer which is at the same level as Xen Dom0 (a,b). In our analysis, we focus on the parameter considering the following cases. Case a: the setup is exactly as shown in the Fig. 2. Case b: the DomUs are removed and the parallel application is executed in Dom0s of all computers. Case c: we compare this performance to the performance on a linux machine with no Xen hypervisor installed on it (in the rest of the paper, we call it as native linux). Case d: situation is as in Case a, but the guest operating systems are in the process of live migration. This way, we present a comparison and analyze the performance issues in the Xen hypervisor for high performance computing systems.

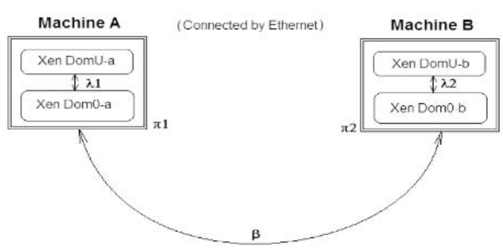


Fig. 2. Machines A and B are connected using 100 MBPS Ethernet. Symbols β, λ, and π represent the following delays. β – time spent on the wire (includes the router). λ – time for message to route from the guest OS to Xen domain0 and π – time for message to get on to the wire from Xen domain0 (includes any network buffers etc).

4. Performance Evaluation

We analyze the performance of the Xen hypervisor under two conditions: First, we conduct the experiments as described in Section 3. This helps us to create a baseline for the appraisal of the Xen hypervisor to be used in the high performance computing systems. Second, we conduct similar experiments however, this time we have the live migration in effect. We do this to analyze the suitability of using the hypervisor mechanism to create an autonomous load balancer in the high performance computing systems, which would adjust the load on a cluster evenly depending upon the current requirements of the applications running on it.

4.1. Performance evaluation without migration

As described in Section 2, we benchmark the performance of Native Linux-Native Linux, DomU-DomU, and Dom0-Dom0 using the Intel's benchmarking suite. The performance curves for single transfer, parallel transfer, and collective classes are shown in figures 8, 9, and 10 respectively. We notice that the performance of Dom0-Dom0 is the best among all for all classes of benchmarks. Next follows Native Linux-Native Linux and lastly DomU-DomU.

An interesting observation is that the performance of Dom0-Dom0 is the best and not of Native Linux. While one may attribute this affect to the larger number of processes in the native linux when compared to Dom0, but we found that in reality this is not the case. We found that, with the original scheduling algorithm on native linux, whenever there is a pause in the communication between the MPI processes, the operating system schedules other active processes. Due to the setup time to schedule a process such as context switching etc. the performance suffered. We confirmed this intuition through (a) removing most of the active processes in the native linux and (b) modifying the scheduler by giving highest priority to the MPI processes and avoid immediate re-scheduling. In the first case, we saw no improvement in the performance but in the second case, we saw that both performances match. With the modified scheduler, the performance is always comparable to Dom0. This effect is shown in Fig. 3 (c) and (d) for the PingPing and PingPong functionalities. So the take home message here is that, in the guest operating systems, and the Xen hypervisor by giving high priorities to processes running MPI jobs, the performance can be improved.

Next, we observe that in the single transfer class, the performance of the DomU-DomU without counter traffic (PingPong case) is close to the performance of Dom0Dom0 and native linux. However, with counter traffic on the network (PingPing case) and in the case of parallel transfer class (Fig. 4) the increase in the gap in the performance of the DomU-DomU case w.r.t. the Dom0-Dom0 and native linux increases almost linearly with the size of the messages. Similar effect is seen in all cases in the collective class (Fig.5) also, whenever there is heavy traffic on the network (Allgather, Allreduce, Alltoall, ReduceScatter). However, in cases where the amount of traffic is lesser, as in the case of Reduce, the performance of DomU-DomU is better. In case of Broadcast where only one process broadcasts to others, the performance of DomU-DomU is very close to that of Dom0-Dom0 and native linux.

We followed the operation of the virtualization of the network device in the Xen hypervisor to explain the difference in performance. In Xen, a network device is virtualized in the following way. In order to transmit a packet, a guest operating system enqueues a buffer descriptor on the transmit ring and copies packet header and executes some filter matching rules. The relevant page frames are pinned until the transaction is complete. Similarly, to receive a message, the guest operating system exchanges an unused page frame with Xen hypervisor. Both, receive and send methods, as described above, require many steps (including multiplexing the request). Although, these steps are included in order to ensure security and isolation of the network device, this becomes a cause for the performance gap when there is counter traffic on the network. This statement indicates that the ability to effectively utilize the bidirectional capabilities of the hardware needs to be improved in the hypervisor, because if the send and receive events are fully separated from each other, then they can be processed in parallel because the connecting cable is fully capable of handling bidirectional transfers. This can be clearly seen by comparing the performance of the DomU-DomU to Dom0-Dom0 and native linux. It is evident that in the latter cases, the operating system is able to effectively utilize the network bandwidth.

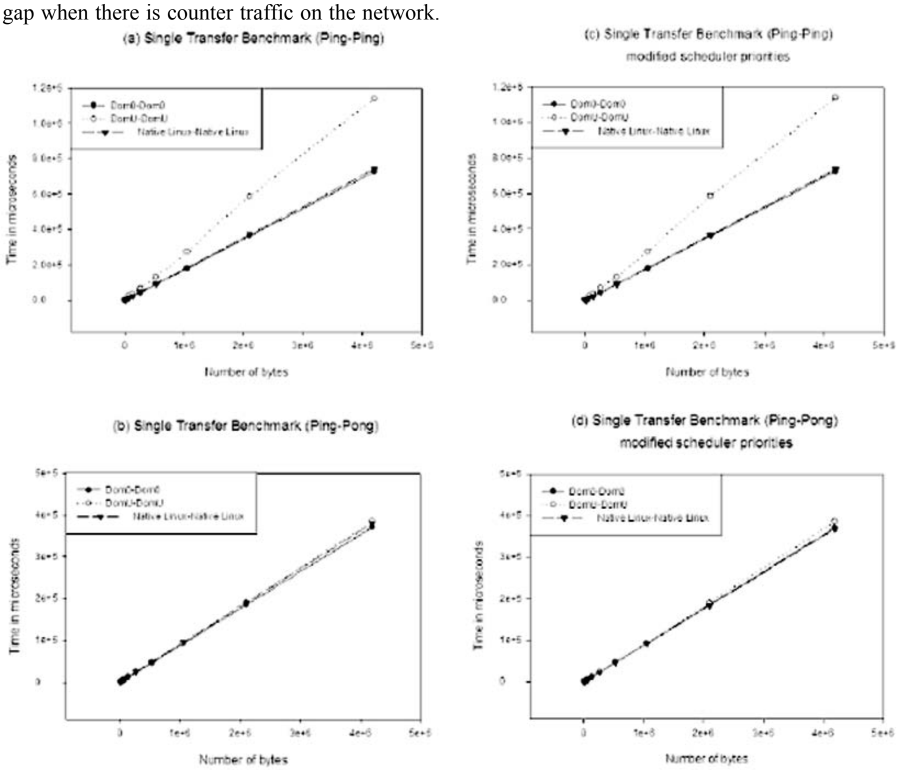


Fig. 3. Performance of Single Transfer Benchmarks

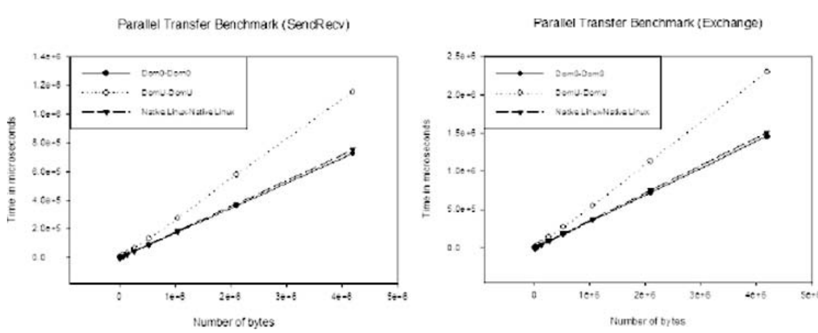


Fig. 4. Performance of Parallel Transfer Benchmarks

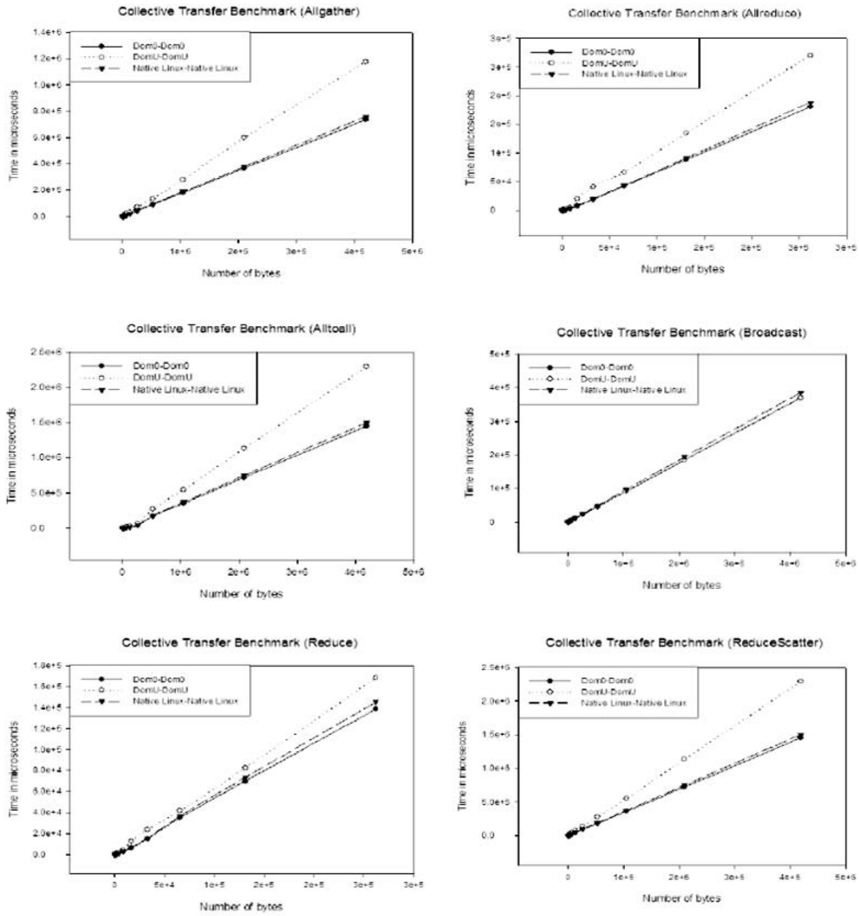


Fig.5. Performance of Collective Transfer Benchmarks

4.2. Performance evaluation with migration in effect

Now we evaluate the performance with migration in effect. We run the benchmarks as before, but during execution we migrate the guest operating systems physically from one machine to another. However, since we found that the behavior of the collective class follows similar patterns as single and parallel benchmarks (as explained in Section 4.1), in this experiment we chose to evaluate only single transfer (Fig.6) and parallel transfer (Fig.7) benchmarks.

The results indicate that live migration of a guest operating system in the case of single transfer class shows a pronounced change around the time when the migration occurs. What becomes more interesting is that, after guest operating systems have been moved to the same domain, the performance does not match with the optimal under that condition. For example, in Fig.6 when we migrate one of the guest operating system from a different machine and place two guest operating systems physically on the same machine, we expect the performance curve to converge with the performance curve of DomU-DomU under same Dom0s. However, we have not seen that pattern. Instead, we see that the performance gets worse. One may argue that since multiple domains are executing on the same machine, the performance degrades. However, this argument does not hold because, in these benchmarking suite we are only measuring the network performance through Sends and Receives and do no computationally intensive work. In the case of parallel transfer, no pronounced effect is seen, but the whole performance curve as a whole gets shifted up, indicating a uniform degradation (Fig.7) in the performance. However in all these observations the actual time to migrate is in the order of milliseconds as stated in the original paper [5], but the behavior after migration is not seen to converge to the ideal. In fact it gets worse than normal.

To understand the reason behind this observation, we studied the flow of message path after migration. We found that, after migration, since the guest operating systems retain their IP addresses and keep all the connections open, the messages are actually being routed through the router. However, they could have been easily routed through the hypervisor itself because after migration the guest operating systems were on the same physical machine. The continuous degradation in performance curve after migration in case of single transfer class of benchmarks is explained through the reasoning we presented in Section 4.1 about the methodology of virtualizing the network device in the Xen hypervisor.

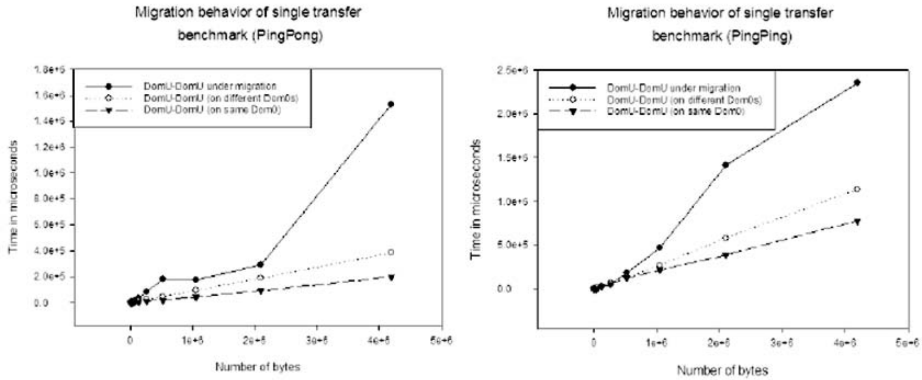


Fig.6. Performance of Single Transfer Benchmarks under migration

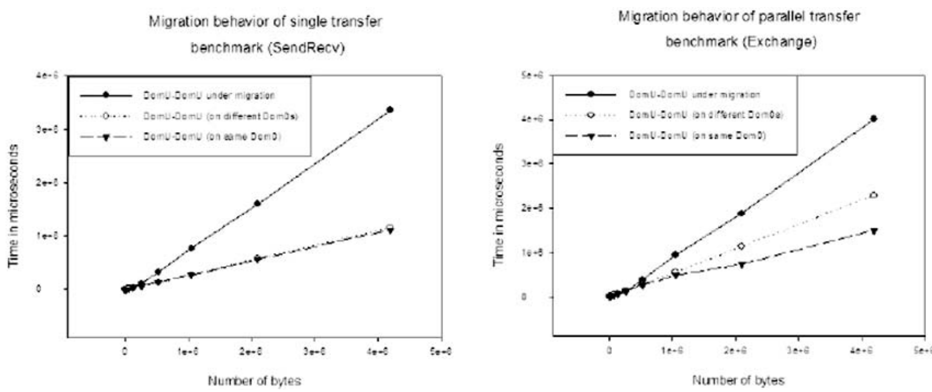


Fig.7. Performance of Parallel Transfer Benchmarks under migration

5. Conclusions

From our study, we see that the performance of the hypervisor under ideal conditions is comparable or better than the native linux, but under some stressful conditions such as live migration and non-optimal network traffic the performance suffers severely. The Xen hypervisor system in its current state is unsuitable for high performance computer systems. However, its inherent qualities of performance isolation and security are very helpful in the high performance computing systems to achieve fault tolerance and uniform load balance. With some modifications to the Xen kernel, which we discuss below, we believe that it can be made very suitable for the high performance computing architectures.

First, the hypervisor should be able to fully utilize the bidirectional capabilities of the network hardware. Second, Xen hypervisor should be able to identify inter-domain and intra-domain messages after migration of the guest operating systems. Third, the scheduler of the guest domains and domain0 should provide high priority to processes executing parallel applications to other processes such as operating systems daemons. Fourth, the hypervisor layer should be made thinner such that it is just able to handle the minimum services required in a cluster environment. What are the minimum required services is a research topic in itself, which is currently under investigation [14].

In the future, we plan to modify the Xen kernel according to the observations we made in this paper to make it suitable for the high performance computing systems. Our immediate goal is to be able to develop an autonomous load balancing system through virtualization techniques for which we plan to use the Xen hypervisor.

References

[1] The Xen Virtual Machine Monitor, Available: http://www.cl.cam.ac.uk/Research/SRG/netos/xen/

This page intentionally blank

[2] B. Clark, T. Deshane, E. Dow, S. Evanchik, M. Finlayson, J. Herne, J. N. Matthews, "Xen and the art of repeated research", USENIX, June 27-July2, 2004.
[3] Xen Users' Manual, Xen v2.0 for x86. Available: http://www.cl.cam.ac.uk/Research/SRG/netos/xen/readmes-2.0/user/user.html
[4] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, A. Warfield, "Xen and the art of virtualization", SOSP 2003
[5] C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I. Pratt, A. Warfield, " Live migration of virtual machines", USENIX 2005
[6] M. Sottile, V. Chandu, D. Bader, "Performance analysis of parallel programs through message-passing graph", IPDPS 2006.
[7] Intel Cluster Toolkit-2.0. Available: http://www.intel.com/cd/software/products/asmona/cluster/clustertoolkit/index.htm
[8] Open SSI, Single System Image for Linux Clusters. Available: http://www.openssi.org
[9] M. Hardt, F. Karlsruhe, "Xen: Experiences and performance measurements". Linux 2005
[10]D. Smith, K. Dukowicz, R. Malone, "Parallel Ocean General Circulation Model", Physica D, vol. 60, pp. 38-61, 1992
[11]MPICH-A Portable Implementation of MPI. Available: http://www-unix.mcs.anl.gov/mpi/mpich1
[12]K. Duda, D. Cheriton, "Borrowed Virtual Time (BVT) scheduling: supporting latency-sensitive threads in a general-purpose scheduler". In Proc. 17th ACM SIGOPS Symposium on Operating System Principles, 1999, vol 33, no. 5, pp 261-276.
[13]Debian-The Universal Operating System. Available: http://www.debian.org
[14]R. Minnich, M. Sottile, S. Choi, E. Hendriks, J. McKie, "Right Weight Kernels: an-off-the-shelf alternative to custom Light-Weight Kernels." Special Issue of the ACM Operating Systems Review Journal 2006.

Interpolation for Super Resolution Imaging

Varsha Hemant Patil
Computer Engineering Department, KKWIEER Nashik-3, India.
Dattatraya S. Bormane
Rajashree Shahu College of Engineering, Pune, India

Abstract-High Resolution (HR) means pixel density within image is high. Along with pleasing picture, the high resolution image offers additional information that may be vital in analyzing image precisely in applications like, military, medical imaging, consumer electronics and so forth. Super resolution image reconstruction is used to restore a high resolution image from several low resolution images. Super resolution image reconstruction is three stage process: Registration, Interpolation and Restoration. In this paper we suggest a wavelet based interpolation that decomposes image into correlation based subspaces and then interpolate each one of them independently. Finally combine these subspaces back to get the high resolution image. We propose it for super resolution imaging along with results to put forth that it produces best results qualitatively analyzed using subjective quality measure. The concepts related to super resolution imaging; interpolation and wavelet are covered as background theory.

Keywords: *Image Interpolation, Super-Resolution and Wavelet.*

I. INTRODUCTION

In most electronic imaging applications, images with high resolution are desired and often required. Pixel density within high resolution image is high, and therefore high resolution image can tender supplementary details that may be critical in various applications. High resolution medical images are very helpful for a doctor to make correct diagnosis. In case of satellite images, it may be easy to distinguish an object for similar ones using high resolution image. Also the performance of pattern recognition in computer vision can be improved if an HR image is provided. In most of these applications, zooming of image is achieved by using interpolation of region of interest. Also to correct the aspect ratio, interpolation is used to increase the resolution of image. As interpolation itself has limitations, discovery of approach to boost the current resolution level is needed. Super Resolution (SR) image reconstruction is currently providing the solution to this problem and is one of the most spotlighted research areas. It can overcome the natural resolution constraint of the imaging system and perk up the performance of most digital image processing applications.

A. Super Resolution Image Reconstruction

Resolution of image is dependent on the resolution of image acquisition device. Since 1970s, CCD and CMOS sensors have been widely used to capture digital images. The current resolution level and consumer price are not satisfying current and future demands of imaging applications. The most direct solution is to reduce pixel size, that is, increase number of pixels per unit area by sensor manufacturing technique. But as pixel size reduces the amount of light available also decreases and it generates shot noise that in turn degrades the image quality. Another approach for increasing spatial resolution is to increase the chip size, which leads to increase in capacitance. Since large capacitance would make it difficult to speed up the charge transfer rate, this approach is not feasible. For these hardware based solutions, the limitations of manufacturing technology of high precision optics and image sensors are important concerns from commercial point of view. Hence a software approach towards increasing the spatial resolution is required to overcome these limitations. Software approach based on signal processing approach is referred as super resolution imaging. The most important advantage is that it may cost less and existing low resolution imaging systems can still be used.

Super-resolution (SR) restoration aims to solve the problem: given a set of observed images, estimate an image at a higher resolution than is present in any of the individual images. Researchers have developed variety of super resolution approaches in last decade. The basic premise for increasing the spatial resolution in super resolution technique is availability of multiple low resolution images captured from the same scene. These multiple low resolution images represent different looks of the same scene. The super resolution image reconstruction is three stage process: Registration, Interpolation and Restoration. Registration refers to motion compensation that is used to map the motion from all available low resolution frames to a common reference frame. The motion field can be modeled in terms of motion vectors or as affined transformations. The assumption is that all pixels from available low resolution frames can be mapped back onto the reference frame. Next, in order to obtain a uniformly spaced up-sampled image, interpolation onto a uniform sampling grid is done. This is to map the motion compensated pixels onto a super resolution grid. The third stage restoration is used to remove the sensor and optical blurring in resultant up-sampled image. There are methods developed for pixel registration based on pixel shift, motion blur, phase correlation or photometric cue, training images etc. [28-32]

In this paper, we present a wavelet based interpolation scheme for super resolution process, which can even be used to zoom the image. This section has covered concepts related to super resolution imaging. In section II, we present a brief literature review of interpolation and its techniques. In section III, the problem domain is formulated. In section IV we present the solution guidelines on which the proposed technique is based along with supporting wavelet features. We pick a few applications such as medical imaging and natural images and show results of proposed technique in section V. Conclusions are presented in section VI.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 483–489.

II. INTERPOLATION

Interpolation also known as re-sampling is an imaging method to increase (or decrease) the number of pixels in a digital image. Interpolation transforms a discrete matrix into a continuous image. It is process of fitting the data with a continuous function and re-samples the function at finer intervals as per need. Hence interpolation is the process by which we estimate an image value at a location in between image pixels. For example, if you resize an image so it contains more pixels than it did originally, it calculates values for the additional pixels through interpolation.

A. Existing Interpolation Techniques

Image interpolation has been a key branch of Image Processing and is used for zooming or for increasing the resolution of the image. Much work has been done in this regard. Various techniques have been developed. Some digital cameras use interpolation to produce a larger image than the sensor captured or to create digital zoom. Virtually all image editing software support one or more methods of interpolation. There are many interpolation methods: Nearest neighbor interpolation, Simple Replication, Bilinear interpolation, Bicubic interpolation, Quadratic, Lagrange, B-spline, sinc, Gaussian among others. The interpolation methods all work in a fundamentally similar way. The difference among various approaches lies in the interpolation model chosen [1,2]. In each case, to determine the value for an interpolated pixel, the point in the input image is located that the output pixel corresponds to. Then a value to the output pixel is assigned by computing a weighted average of some set of pixels in the vicinity of the point. The weightings are based on the distance each pixel is from the point. The methods also differ in the set of pixels that are considered. For nearest neighbor interpolation, the output pixel is assigned the value of the pixel that the point falls within. No other pixels are considered. For bilinear interpolation, the output pixel value is a weighted average of pixels in the nearest 2-by-2 neighborhood. For bicubic interpolation, the output pixel value is a weighted average of pixels in the nearest 4-by-4 neighborhood. The number of pixels considered affects the complexity of the computation. Therefore the bilinear method takes longer time than nearest neighbor interpolation, and the bicubic method takes longer than bilinear. However, the greater the number of pixels considered, the more accurate the effect is, so there is a trade-off between processing time and quality. Let us see details of few basic techniques of interpolation.

Nearest Neighbor Interpolation

Nearest neighbor interpolation is the simplest method and basically makes the pixels bigger. The color of a pixel in the new image is the color of the nearest pixel of the original image. If you enlarge 200%, one pixel will be enlarged to a 2 x 2 area of 4 pixels with the same color as the original pixel. Most image viewing and editing software use this type of interpolation to enlarge a digital image for the purpose of closer examination because it does not change the color information of the image and does not introduce any anti-aliasing. For the same reason, it is not suitable to enlarge photographic images because it increases the visibility of blocking artifacts.

Bilinear Interpolation

Bilinear Interpolation determines the value of a new pixel based on a weighted average of the 4 pixels in the nearest 2 x - 2 neighborhood of the pixel in the original image. The averaging has an anti-aliasing effect and therefore produces relatively smooth edges with hardly any artifacts.

Bicubic interpolation

Bicubic interpolation is more sophisticated and produces smoother edges than bilinear interpolation. Here, a new pixel is a bicubic function using 16 pixels in the nearest 4 x 4 neighborhood of the pixel in the original image. This is the method most commonly used by image editing software, printer drivers and many digital cameras for re-sampling images. Adobe Photoshop CS offers two variants of the bicubic interpolation method: bicubic smoother and bicubic sharper.

Fractal interpolation

Fractal interpolation is mainly useful for extreme enlargements (for large prints) as it retains the shape of things more accurately with cleaner, sharper edges and less halos and blurring around the edges than bicubic interpolation would do. Genuine Fractals Pro from The Altamira Group includes it.

An interpolation is resizing the image; both increase and reduce the resolution. When the size of an image is reduced, some of the original pixels are loosed because there are fewer pixels in the output image. Aliasing that occurs as a result of size reduction normally appears as "stair-step" patterns (especially in high-contrast images, or as moiré (ripple effect) patterns in the output image. When either bilinear or bicubic as the interpolation method is used the tool using it automatically applies a lowpass filter to the image before interpolation to limit the impact of aliasing on the output image.

B. Interpolation for Super Resolution

The zooming algorithm is the process to enlarge a picture preserving the details. If we have two or more frame of the same scene, it possible to obtain a sharpen image, using the super resolution techniques. Usually, some Digital Cameras allow choosing the best frame between the numerous frames acquired simultaneously. With the super resolution it is possible to merge such Low-Resolution (LR) frames to obtain a new enhanced picture, where the relative misalignment between successive frames of the same scene allows recovering more high frequency details. In this way the physical limits of the acquisition system are substantially bypassed by properly using more than one frame.

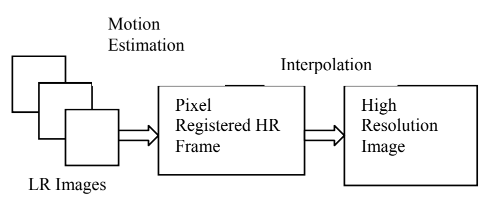


Figure 1. Stages of Super resolution Image Reconstruction

The first stage of the super resolution process is the motion estimation and alignment of the frame. Then the frame is zoomed using some well-known interpolation technique, and then the information relative to the corresponding pixels is merged. A schematic description process is given in figure 1. We suggest interpolation technique for second phase of process that allows to sensibly improving the visual quality of the image.

III. PROBLEM FORMULATION

Though plenty of interpolation techniques, each one of them suffer from various drawbacks. Linear interpolation tries to fit a straight line between two points that leads to blurred image. The line is drawn by linearly interpolating unknown points between grid points. Pixel replication copies the neighboring pixel to empty location that tend s to produce blocky image. The simplest technique nearest neighbor takes values point same as that of grid point closest to it. It introduces strong aliasing and blurring effect. Spline and Sinc interpolation techniques try to reduce these extremities. Spline interpolation is inherently a smoothing operation, while Sinc produces ripples in image. A technique proposed in [3] is Bayesian approach for zooming. A primary focus of developers/researchers is treating interpolation as an approximation problem subject to some standard continuity conditions. Even though there has been a spate of development what is often overlooked is that the quality of the interpolated image is judged by the way it is perceived using PSNR. A particular interpolation scheme may perform quite well for an image if it contains object of nearly similar texture. Profound study report has been presented in [4]. It is stated that all interpolation methods smooth the image more or less. Images with sharp edged details and high local contrast are more affected by interpolation than others. In [4], researcher has presented the comparison accomplished by spatial and Fourier analysis of the kernel functions, visual quality assessment, quantitative interpolation error determination, computational complexity analysis and run time measurement based on representative applications and clinical images. Also key features of each method are stressed.

The stages registration and interpolation of super resolution image reconstruction are based on fact that low resolution images are sub-sampled as well as mis-registered with subpixel shifts. If images are shifted by integer amount then each image contains the same information and thus there is no new information that can be used. However, if the images have sub-pixel shifts and if aliasing is present then each image has different shifts. New information is therefore contained in each low resolution image, and thus can be exploited to obtain high resolution image. Hence interpolation is used as the process of computing samples of high resolution image from these shifted low resolution images either in spatial domain or in frequency domain.

IV. SOLVING THE PROBLEM

The basic idea behind proposed technique is to avoid the constraints of these existing interpolation methods. The analysis of reasons introducing unwanted effects is done. From study few conclusions are drawn.

- From sampling theory shows that the scanning of continuous image yields infinite repetitions of its continuous spectrum in Fourier domain, which do not overlap since the Nyquist criterion is satisfied. Isf this is so, and only the original image can be constructed perfectly from its samples by multiplication of an appropriate rectangular prism in Fourier domain.
- Sampling the interpolated image is equivalent to interpolating the image with a sampled interpolation.
- The ideal interpolation for point says A (x, y) should be 1-D interpolation in x-direction first by four one-dimensional interpolations. They should be then used for the final 1-D interpolation in y-direction.
- The transfer function of the ideal interpolator is constant; it is one in the pass-band and zero in stop-band.
- The 1-D interpolation equals the multiplication with a rect function the Fourier domain and can be realized by a convolution with the sinc function in the spatial domain.
- Although the sinc function provides an exact reconstruction of image it is spatially unlimited, hence truncated and windowed.
- Truncation of the ideal interpolator produces ringing effect n frequency domain as considerable amount of energy is discarded.

Hence to overcome the drawbacks of existing interpolation techniques few points are to taken care of:

- Edges play an important role in perception, due emphasis is to be given while processing them. If the perceptually relevant features are to be extracted and interpolated separately, perceptual artifacts, mostly at object boundaries can be avoided.
- Requirement is to take care of neighborhood structure especially at the grid points at boundaries. Most of the interpolation techniques are based on approximations with respect to neighboring pixels that leads to blur or blocking effect. This is effect is more prominent when the neighboring pixels are not correlated.
- The technique is needed that would perceptually group the features and treat them individually as per their perceptual relevance.
- Hence it is needed to decompose the image into appropriate correlated components and interpolate them separately and then to transform back to the image.

Also from study of wavelets few thing are noted:

- Recently wavelet has emerged as a promising tool for Signal and Image Processing
- Wavelets are mathematical functions that slice up data into dissimilar frequency components, which in turn helps to learn each component with a resolution matched to its level.
- Using wavelet analysis, we can use approximating functions that are contained neatly in finite domains. Wavelets are well-suited for approximating data with sharp discontinuities.

- It can decompose a digital image into some frequency sub-images, each represented with proportional frequency resolution. The resulting band-pass representation provides the solution space of many image processing problems.
- Wavelet coefficients decay across scales and can be calculated to estimate wavelet coefficients at finer level. It might help to avoid elimination of high frequency components hence use of wavelet may not suffer from smoothing effects. It may lead to produce images with shaper and less blocking artifacts.
- Wavelet tool helps to clearly classify the neighborhood structure. Wavelet tool decomposes image into approximation, vertical details, horizontal details and diagonal details.

All these observations are drawn from the study of various interpolation techniques and study of wavelet; and also design considerations, results and applications of them.

V. THE SOLUTION: THE PROPOSED TECHNIQUE

The new algorithm is based on all considerations listed in earlier sections. The key idea is to perform correlation controlled, structure preserving interpolation. The technique is described for gray scale images. The generalization to color images can be done based on it.

The algorithm works in three stages:

1. Decompose into subspaces based on structural correlation
2. Interpolate each subspace individually
3. Integrate these subspaces into image

Let us see details of each of these phases.

Phase 1: Firstly decompose image into subspaces. This decomposition is based on structural correlation. The correlation factor we used is frequency component to preserve structural relationship. Aim is to test few mutually exclusive conditions for pixel's luminance value with respect to its surrounding pixels to find the structural relationship. Test should correlate the pixels and accordingly group them. The image is decomposed horizontal, vertical, approximate and diagonal components. We use wavelet (db4) for this decomposition. Further we analyze diagonal component to logically decompose it into two diagonals SW-NE and NW-SE to interpolate them separately in step 2.

Phase 2: Each subspace is interpolated independently.

Case 1: Approximate components- for pixel say P to be interpolated. Compute its value using four or more of its surrounding pixels as:

$P=(A + B + C + D)/4;$

Case 2: Diagonal Component-consider both the diagonals separately. For diagonal in the SW-NE direction, the value (B + C)/2 is assigned to pixel P. For diagonal in the NW-SE direction| the value (A + B)/2 is assigned to pixel P.

Case 3: Horizontal component- Here P is assigned value depending on the location of P either (A + B)/2 or (C + D)/2.

Case 4: Vertical Component- Similar to step 3, P is assigned value depending on the location of P either (A + C)/2 or (B + D)/2.

Pixel structure assumed is shown below in figure 2.

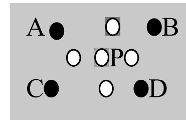


Figure 2: Pixel Structure

3. Wavelet is used again to convert these subspaces back to image domain.

Later the interpolation is often used to mix together the multiple images. Blend fractions (threshold) and (1 - threshold) are used in a weighted average of each component of each pixel:

$$P = (1 - \text{threshold})*p1 + \text{threshold}*p2$$

Typically threshold is a number in the range 0.0 to 1.0. This is commonly used to interpolate two images. Threshold ranges between the intervals 0.0 to 1.0. Threshold we have computed by many trails through experiments on range of images.

VII. RESULTS

The image in figure 3 is a original images and is a image generated during mammography [12,13]. The image has two suspicious patches marked with white colored arrows. The image in figures 5 and 7 are images at upper patch and lower patch magnified of original image respectively. The image in figure 4 is the image with high resolution increased. Figures 6 and 8 display the image with images at upper patch and lower patch magnified for resultant images respectively. All these images are generated as a result of our technique implemented in MatLab7.1. From the results, it is found that for the original low resolution image, if we try to magnify the region of interest the blocking effects are observed. With our super resolution algorithm these effects are eliminated. Moreover, the proposed scheme obtains the superior image quality about the edges. Also more iteration can be applied to increase the resolution further till it gives adequate details to the physician. From the experimental results, we found that the proposed technique is useful for medical as well as natural and computer generated images. Image quality is one of the important factors in image processing applications and it is a critical aspect in medical image processing.

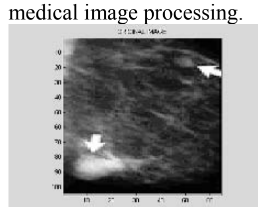


Figure 3 Original Image with two suspicious patches

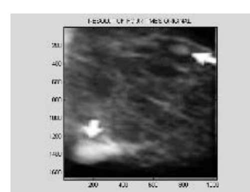


Figure 4: Resultant Image

For image quality measure there are two commonly used techniques: Objective evaluation and Subjective evaluation.

We have used subjective evaluation criteria for image quality. We found it the most suitable for said application.

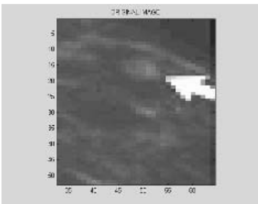


Figure 5: Original Image with upper suspicious patch magnified

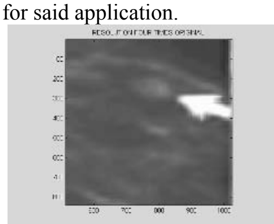


Figure 6: Lower suspicious patch magnified for Resultant Image

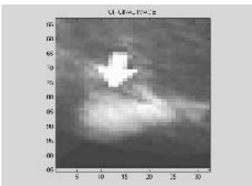


Figure 7: Original Image with lower suspicious patch magnified

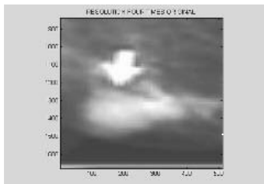


Figure 8:Upper suspicious patch magnified for Resultant Image

It is based on Human Visual System, which allow a better correlation with the response of the human observer. Here the original image and the region of interest are shown with high resolution.

Because characteristics of the edges in a digital medical image are to be reserved for many scales of resolution and edges are always important for human vision, our wavelet based technique preserves the local edge structure to prevent the blurring and blocking effects in reconstructed high-resolution image.

Table1: Subjective Test

Image	Grade score assigned by 90% of Observers					
	Bicubic	Pixel replica	Nearest Neigh.	Bilinear	Fractal	**Proposed**
Lena	2	3	3	3	2	**1**
Woman	2	3	3	3	2	**1**
Mandrill	3	4	3	3	3	**2**
Mammogram	2	3	3	3	2	**1**
Ashiwarya	2	3	3	3	2	**1**
Text	3	4	4	4	3	**2**

We conducted experiment by taking 40 doctors as observers. The five sets with two images per set were presented to the observers, one original image and second the reconstructed HR image with different increasing factors. They evaluated image quality of both images with grades as 1== Excellent, 2== Good, 3== Fair, 4==Poor, 5== Bad. Grades received were proportional to the increase factor of resolution. For images reconstructed with increasing resolution by factor four received the grade 1, that is, excellent from 90% of the observers (Table 1). Figures 9 show a bone image and its separated components. Figure 10 displays the resultant bone image and ROC. Figures 11,12 and 13 demonstrate proposed algorithm's results for few natural images.

VII. CONCLUSION

In this paper we have proposed a new technique for interpolation. We have analyzed it problem from the wavelet point of view. From experimental results it is observed that the proposed method is competitive or rather better in quality and efficiency with the existing interpolation methods. The proposed method has been compared to bicubic interpolation, pixel replication and many others. We are currently working on two extensions of our work: incorporating more details in the image by super resolving the image to higher factor using further decomposition of wavelet components. And extending our method to work better for situations where more than one low-resolution image is available. However we feel that wavelet based super resolution technique has the potential to make a significant impact on medical imaging analysis.

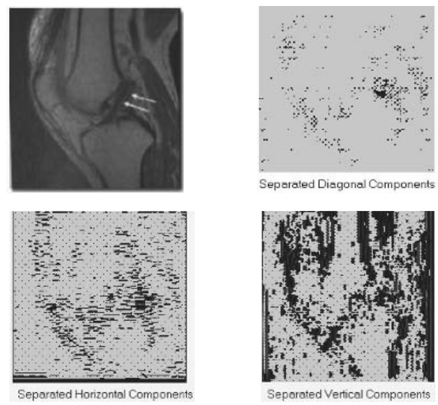


Figure 9: Bone Image and its separated Components.

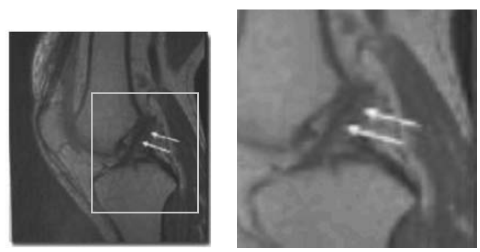


Figure 10: Bone Image and ROC

Figure 11: Original Aishawarya image and resultant HR image

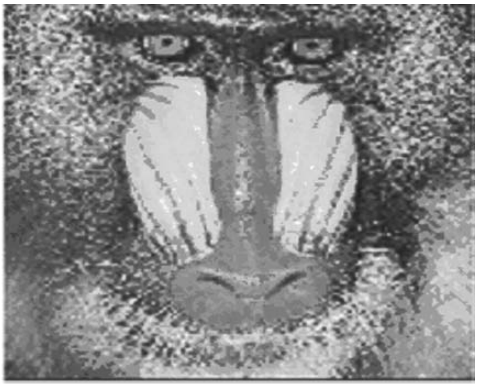


Figure 12: Low Resolution Mandrill Image

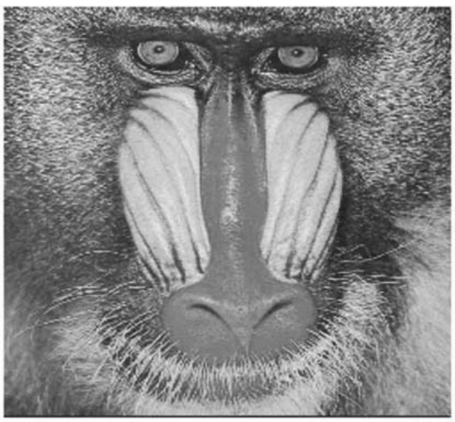


Figure 13: Resultant High Resolution Mandrill Image

REFERENCES

[1] H. S. Hou and Andrews "Cubic spiles for image interpolation and digital filtering,", IEEE trans. On accost, speech and signal processing, vol 6, no 6, pp 508-517, 1978.

[2] R G Keys, " Cubic convolutions interpolation for digital image processing", ISSS Tran. On accost, speech and signal processing, vol 6, no 6, pp 1153-1160, 1981.

[3] Richard Schutz and Stevenson, "Bayesain approach to image expansion for improved definition", IEEE Trans. On Image Processing, pp-234-241, May 1994.

[4] T.M. Lehmann, C. Gonner, K. Spitzer, Survey: Interpolation Methods in Medical Image Processing, IEEE Transactions on Medical Imaging, Vol.18, No.11, November 1999;

[5] E. Maeland, On the comparison of interpolation methods, IEEE Transactions on Medical Imaging, Vol.MI-7, pp.213-217, 1988;

[6] D.M. Monro, P.D. Wakefield, Zooming with Implicit Fractals, Proceedings of International Conference on Image Processing ICIP97, Vol.1, pp.913-916 1997;

[7] W.T. Freeman and E.C. Pasztor and O.T. Carmichael. Learning low level vision. International Journal of Computer Vision, 40(1):25–47,2000.

[8] W.T. Freeman, T.R. Jones, and E.C. Pasztor. Example based super resolution. IEEE Computer Graphics and Applications, 22(2):56–65, 2002.

[9] Gunturk, Batur, Altunbasak, Hayes and Mersereau, Eigenface based super resolution for face recognition, IEEE, II-845- II848, IEEE ICIP 2002.

[10] Anil K Jain, Fundamentals of Digital Image Processing, Prentice Hall

[11] http://www.aicr.org.uk/links.stm

[12] http://www.breastcancer.org

[13] K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images.". IEEE Trans. Image processing, vol.4,no.3, Mar 1995, pages 285-295

[14] K.P. Hong, J.K. Paik, H.j. Kim and C H lee, " An edge preserving image interpolation system for a digital camcorder", IEEE Tran.Consumer Electronics, Vol 42, no 3, Aug 1996, Pages 279-284.

[15] A M Darwish, MS Bedair, and S I Saheen, " Adaptive resampling algorithm for image zooming.", IEE Proc-Vis. Image Signal Processs, vol, 144, No 4, aug 1997, pages 207-212.

[16] M. Antonini, M Barlaud, P Mathieu, and I daubechies, "Image coding using wavelet transform", IEEE Trans. Image processing, vol.1, no.2, Apr 1992, pages 205-220.

[17] T. Chang and C J kao, "Texture analysis and classfication with tree structured wavelet transform", Trans. Image processing, vol.2,no.4, Oct 1993, pages 429-441.

[18] Averbuch, D Lazar, and m Israeli, " Image Compression using wavelet transform and multi resolution decompositions", Trans. Image processing, vol.5,no.1, Jan 1996, pages 4-15.

[19] V. R. Algazi, G. E. Ford, and R. Potharlanka, "Directional interpolation of images based on visual properties and rank order filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, 1991, pp.3005–3008.

[20] S. W. Lee and J. K. Paik, "Image interpolation using adaptive fast B-spline filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, 1993, pp. 177–180.

[21] J. E. Adams Jr, "Interactions between color plane interpolation and other image processing functions in electronic photography," *Proc. SPIE*, vol. 2416, pp. 144–151, 1995.

[22] S. Carrato, G. Ramponi, and S. Marsi, "A simple edgesensitive image interpolation filter," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1996, pp. 711–714.

[23] B. Ayazifar and J. S. Lim, "Pel-adaptive model-based interpolation of spatially subsampled images," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 3, 1992, pp. 181–184.

[24] B. S. Morse and D. Schwartzwald, "Isophote-based interpolation," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1998, pp. 227–231.

[25] K. Ratakonda and N. Ahuja, "POCS based adaptive image magnification," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1998, pp.203–207.

[26] D. Calle and A. Montanvert, "Superresolution inducing of an image," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1998, pp. 232–235.

[27] A. J. Patti, M. I. Sezan, and A. M. Tekalp, \High resolution image reconstruction from a low-resolution image sequence in the presence of time-varying motion blur". Proc. of the IEEE International Conference on Image Processing, vol. I, 1994.

[28] Eric Kaltenbacher, Russell C. Hardie, \High resolution image reconstruction using multiple, low resolution, aliased frames". SPIE, vol. 2751, 1996.

[29] Mohammad S. Alam, John G. Bognar, Russell C. Hardie, Brian J. Yasuda, \High resolution image reconstruction using multiple, randomly shifted, low resolution, aliased frames". SPIE, vol. 3063, 1997.

[30] Hassan Foroosh, Josaine B. Zerubia, \Extension of phase correlation to subpixel registration". IEEE Trans. on Image Processing, vol. 11, no. 3, March 2002. 83

[31] Hassan Foroosh, Scott Hoge, \Motion information in the phase domain". in Video registration (editors M. Shah and R. Kumar), vol. 5 of Kluwer

This page intentionally blank

Intl Series in Video Computing, ch. 3, Kluwer Academic Publishers, May 2003.

[32] Dirk Robinson, Peyman Milanfar, \Fundamental performance limits in image registration". IEEE Trans. on Image Processing, vol. 13, no. 9, Sep. 2004.

[33] Nhat Nguyen, Peyman Milanfar, and Gene Golub, \A computationally e±cient superresolution image reconstruction algorithm".

[34] Michael K. Ng and Nirman K. Bose, \Mathematical analysis of super-resolution methodology". IEEE Signal Processing Magazine, May 2003.

[35] Zhouchen Lin, Heung-Yeung Shum, \Fundamental limits of superresolution algo- rithms under local translation". IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 26, no. 1, Jan. 2004.

A Content Management Implementation at Intercollege

Vasso Stylianou, Andreas Savva
Department of Computer Science, School of Sciences and Engineering, Intercollege
P.O. Box 24005, 1700 Nicosia, Cyprus. Tel: +357-22841647, Fax: +357-22357481
{stylianou.v, savva.a}@intercollege.ac.cy

***Abstract*-This paper presents in summary the implementation of a Web Content Management System (WCMS) at Intercollege. The system was developed to be used by the Research Office of Intercollege. As a Content Management System (CMS) the system presents a tool that can essentially by used to manage content, which materializes here as research activities and knowledge obtained, in various ways, so that it can be quickly and easily published and accessed. The purpose of this system, just like other CMSs, was to record, analyze, and be able to re-use accumulating information on research activities of Intercollege employees.**

Keywords: Knowledge Management, Content Management Systems, Web Content Management

I. A Short Introduction to Knowledge Management and Content Management Systems

There may not be a universal definition of Knowledge Management (KM), but in a broader context this may be defined as the process through which organizations generate value from their intellectual and knowledge-based assets. Most often, generating value from such assets involves sharing among employees, departments and even with other companies in an effort to devise best practices. In other words, it is your firm's system for capturing, managing and re-using the knowledge that resides in electronic documents on your network or, more important, the tacit knowledge[1] that is in your employees' heads.

The use of KM in the business era is not entirely new. The initial tasks of knowledge management which include the collection and storage of knowledge in databases are dated back to the Industrial age when various organizations had the need to archive as much descriptive information as possible about their business processes. What is considered relatively new is the management of knowledge using the aid of information technology [2]. In fact, in our modern knowledge economy, where knowledge is greatly valued, knowledge management is one of the most common catchphrases used by vendors to get the attention of corporate executives.

Knowledge management is characterized by the following four basic activities: 1) Knowledge acquisition, 2) Knowledge analysis, 3) Knowledge preservation, and 4) Knowledge utilization. As it is easily understood, to manage knowledge, first of all knowledge must be acquired either through the process of learning or through the process of identification. Then knowledge can be analyzed, stored and used at any time, on any subject of interest.

Content Management Systems (CMS)

Content Management is the process of organizing, categorizing and structuring knowledge in such a way that it can be archived, published and reused.

A Content Management System is a tool that allows end users to manage this content (knowledge) in various ways, so that it can be quickly and easily published. CMSs incorporate simple features like: editing, content search, version history, approval processes, personalization and localization of content. In addition, CMSs include advanced features like dynamic content generation, metadata and taxonomy categorization, content analysis and content security.

The following five types of CMS are the most common and are widely used nowadays by organizations around the world [1]:

1. Web CMS (WCMS): Help organizations in the content creation, management, and delivery through the Web.
2. Transactional CMS (TCMS): Used in e-commerce transactions.
3. Integrated Document Management Systems (IDMS): Help organizations in managing enterprise documents and content.
4. Learning CMS (LCMS): Used for authoring and publishing web-based learning content.
5. Publication CMS (PCMS): Help organizations in managing their publications.

II. Web Content Management at Intercollege

Intercollege is a tertiary education institution located in Cyprus. The Research Office of Intercollege is responsible for the coordination of all research activities undertaken by its employees locally and internationally. To meet its aims and objectives, Intercollege's Research Office needs to acquire, analyze, preserve and utilize knowledge regarding various projects - including proposed, ongoing and finalized projects - and research-related emails. Such tasks can be handled using the proposed Research Management System, a web-enabled Content Management System. As a web-enabled system and since it will be installed on Intercollege's intranet, this CMS will provide the ability to its end-users to access it securely from anywhere within Intercollege eliminating the need for

[1] Tacit knowledge refers to the knowledge, which cannot be readily or easily written down, primarily because it is based on skills. It is the know-how contained in people's heads [1].

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 491–494.

any software to be installed on their desktop computers. Furthermore, such a system will be more easily maintained and upgraded. Through it the Research Office will be able to rapidly publish a variety of content to interested parties.

The methodology selected for developing the Research Management System was that of Rapid Application Development (RAD) [3] that breaks down activities into four phases being: Requirements planning, User Design, Construction, and Cutover.

The first two phases (Requirements planning and User Design), were completed using the Joint Requirements Planning (JRP) and Joint Application Design (JAD) techniques. Both techniques rely heavily on the cooperation and collaboration of the system developer and the end-users.

During the construction phase, a prototype system was created by the developer to be reviewed by the end-users. Modification and upgrades were then made on the prototype until this was considered to be complete. At the cutover phase - the last phase of the RAD model - the constructed system has undergone thorough testing. The main end-user that cooperated with the developers during the development stages of the project was a senior officer in the research office. Other end-users such as faculty were not involved at this time. Training the remaining end-users, being all interested faculty, was scheduled to follow the institution's decision regarding conversion plans. System implementation was considered to have been completed at this stage.

III. RAD Phases of Project Development

A. *Requirements Planning*

The main two problems identified with the existing system were:

1) Tracking proposed, ongoing, and finalized projects as well as current and future research activity of Intercollege's faculty.
2) Publishing and distributing research-related emails, (received in hundreds every week from local and international agencies), to interested parties at Intercollege. Examples of such research activities are requests for collaborators in big projects, announcements for seminars in specific research areas, etc.

Additional system requirements were identified during requirements' planning. According to these the required system should:

- Be Web-enabled (accessed through a web-browser) so as to be easily accessed, maintained and updated.
- Be fully managed by users with administrative privileges.
- Be accessible from Intercollege's Intranet.
- Allow faculty to submit and manage their projects (Proposed/Ongoing/Finalized) and other research interests.
- Provide the ability to its users to track down a specific project or research interest based on various criteria (Title, Author, Date, etc).
- Collect all research-related emails from the mail servers.
- Allow an administrator to format, categorize and distribute emails to user groups.
- Inform a user upon arrival of an email or project of their interest.
- Be tested for stability and security.

B. *System Design*

In a JAD session with the Research Office the attendees re-examined in detail the requirements laid down during the previous phase.

- The system to be designed had to be a Web-based system, which would be accessible through a browser, from any machine belonging to the Intercollege network. Plans for making the system accessible from the Internet were also considered but were later dropped for security reasons.
- The system should provide the ability to users with administrative privileges to fully manage it; that is, to add, edit or delete various content, to publish content of other users, to add or delete users, to add or delete research interest areas and to update the emails database on demand, i.e., at times other than the scheduled ones.
- Additionally, the system should provide tools that would aid the end-users to submit their projects and research activities and then manage (add/edit/delete) all their details.
- Consequently, users should have the ability to track down projects/research-interests/emails using a search tool resembling those found on Internet search engines.
- With regards to research-related emails, the system should retrieve new emails from the mail servers at scheduled time intervals, store them in its database in a readable format, notify the administrator about their arrival and provide the means for their acceptance/rejection and further distribution (publication) to interested user groups. Once the administrator distributes the mail the users concerned should be notified. Finally, the Research Office required that the system should be fast, stable and above all extremely secure due to the importance of the information being manipulated.

C. *Construction*

The construction phase of the project began by taking under serious consideration the hardware and the software infrastructure of Intercollege.

<u>Hardware Infrastructure:</u>

With the term "hardware infrastructure" we mean all the physical hardware components, which are used to interconnect computers and users. The infrastructure includes all transmission media like telephone lines, network cables, antennas, routers, switches and other devices that control the flow of information through transmission paths.

Due to the fact that the system to be built would acquire, manage and store information, a need for a database server was imperative. Two types of database servers were available at Intercollege's Local Area Network, being a Microsoft SQL

Server 2000 and an Oracle 9 Database Server. An analysis of both servers' hardware and software specifications led us to select Microsoft's SQL Server 2000 for the development of the system.

Since Intercollege's Research Office requested to develop the system as a web-enabled application it was necessary to consider another type of server; the Web Server. This would enable end-users to run the system online using their web-browsers from any host at Intercollege's network. In this case the only server option available at Intercollege was Microsoft's Internet Information Services (IIS 6.0), which runs under Windows 2000 Advanced Server of Windows 2003 Server platforms.

Finally, because the system was to interact with a mail server for the retrieval, categorization and publication of research-related emails, a mail server was needed and this was a Microsoft Exchange Server.

Consequently, the information gathered above made it possible to conclude that the infrastructure available at Intercollege could support the system that was about to be built.

Software Infrastructure:

Having in mind the analysis of Intercollege's hardware infrastructure it was clearly visible that a compatible programming language had to be chosen. Also the language chosen would have to be easy to use, scalable, fast and above all reliable. Therefore, after a systematic comparison of all of the widely-used languages it was decided to use Microsoft's ASP.NET Internet programming language. This would allow the fast development of a stable, scalable, and secure system.

In the figure that follows the collaboration of all hardware and software components of the system can be viewed.

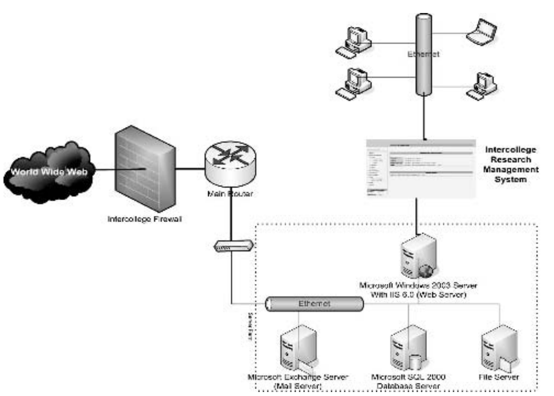


Fig 1. Collaboration of system hardware and software components

Having examined the above enabling technologies and deciding upon the selection of a platform, the next step in the construction phase was to elaborate on the entities comprising the system and their attributes.

Entities of the System:

The Research Office required that the online Web Forms should keep the design of the paper forms used to hand out to researchers. The collection and examination of all of the data that appears on these forms let to the construction of the following Entity Relationship Model of our knowledge base.

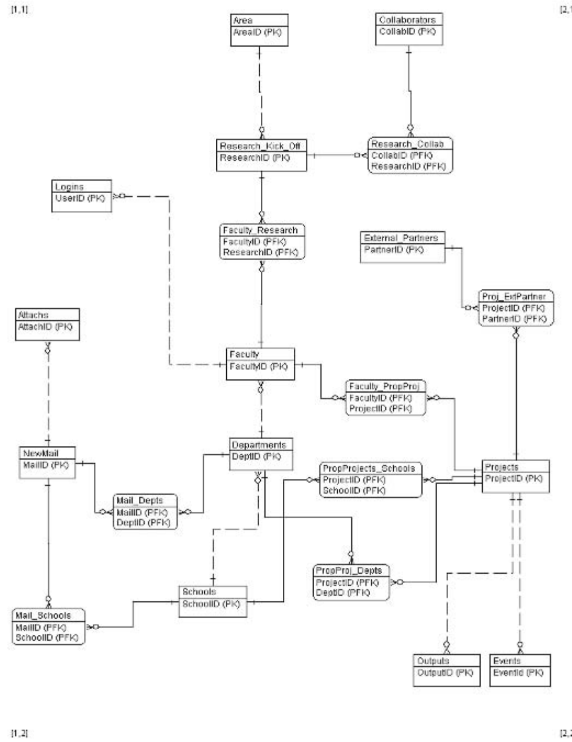


Fig. 2. System's Entity Relationship Diagram (ERD)

Coding the System:

As mentioned earlier, Microsoft's ASP.NET programming language was selected in order to develop the system. ASP.NET is a server-side language – that is, it runs on the server - in which the code is not included within the HTML. The syntax used for ASP.NET language was VBScript. Numerous classes, libraries, procedures and functions in VBScript were created in order to provide the desired output. Some other data structures included in Visual Studio.NET 2003 that provided common functionality and which suited our needs, were used in order to speed up development. Additionally SQL (Structured Query Language) was required for implementing the database schema.

D. The Cutover phase

During the Cutover phase, the last phase of a project, the constructed system went through thorough testing by the development team. Real data such as project proposals and research-related material was obtained and used as testing aids to measure the correctness and usability of the system.

Additionally, at the pre-implementation phase the system was tested against various emails with different content such as images, executable files, etc. to ensure that mails arriving from research-related agencies would be available to the end-users. Then the system was installed on a Web Server dedicated for testing purposes at Intercollege's network, so as to be thoroughly tested for a period of time by the Research Office.

IV. Conclusions

Systems that collect, manage, and distribute knowledge to information workers within an organization are commonly called Knowledge Management Systems (KMS). It should be expected that many of these KMS are custom-built since they need to suit the needs and requirements of specific applications and organizations.

This paper presented a case study on the development of a Web Content Management System at Intercollege. The system under construction being a Research Management System was required to acquire, analyze, preserve and utilize knowledge regarding various projects - including proposed, ongoing and finalized projects - and research-related emails. The methodology followed and a description of all the decisions that needed to be made as well as the overall work that was completed, might prove useful to others with similar requirements that would call for the development of a Web Content Management System.

Acknowledgment

The authors would like to acknowledge the programming efforts of Mr Nicholas Loulloudes, directed towards the completion of an undergraduate final year project in Computer Science.

References and Bibliography

[1] Wikipedia, "Content Management System", June 2005, http://en.wikipedia.org/wiki/Content_management_system

[2] C. Figallo, N. Rhine, *Building the Knowledge Management Network, Best Practices, Tools and Techniques for Putting Conversation to Work, Canada,* John Wiley & Sons, 2002, ISBN 0-471-21549-X, pp. 3-14

[3] H. van Vliet, *Software Engineering, Principles and Practice,* 2nd edition, New York, John Wiley & Sons, 2002, ISBN 0-471-97508-7, pp.57-58

[4] T. Tobin, "The Top 4 Reasons you Need KM – Today", *A Supplement of KMWorld Magazine, Best Practices in Enterprise Knowledge Management*, Vol. IV, volume 13, Issue 10, November/December 2004

[5] I. Watson, *Applying Knowledge Management, Techniques for Building Corporate Memories San Francisco,* Morgan Kaufmann, 2003, ISBN 1-55860-760-9, pp. 7-10

[6] W3Schools, "Introduction to JavaScript", http://www.w3schools.com/js/js_intro.asp

[7] M. J. Rosenberg, *E-Learning – Strategies for Delivering Knowledge in the Digital Age,* McGraw-Hill Companies, Inc, 2001, ISBN 0-07-13628-1, pp.63-72

[8] M. Alavi, "Knowledge Management and Knowledge Management Systems", http://www.mbs.umd.edu/is/malavi/icis-97-KMS/sld001.htm

[9] M. Santozus, J. Surmacz, "The ABC of Knowledge Management", May 2001, http://www.cio.com/research/knowledge/edit/kmabcs.html#what

[10] K. Kendall, J. Kendall, *Systems Analysis and Design,* 5th edition, New Jersey, Prentice Hall International, Inc., 2002, ISBN 0-13-042365-3, pp. 6, 10-16

[11] Microsoft Corporation, "Active Server Pages", 2005, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnanchor/html/activeservpages.asp

[12] J. Crowely, "Beginning Active Server Pages – Introduction", 2004 http://www.developerfusion.com/show/1010/1/

[13] Microsoft Corporation, "Technology Overview", 2005, http://msdn.microsoft.com/netframework/technologyinfo/overview/default.aspx

[14] Microsoft Corporation, "Why ASP.NET? Writing dynamic, high-performance Web applications has never been easier", 2005, http://www.asp.net/whitepaper/whyaspnet.aspx?tabindex=0&tabid=1

[15] J. Myers, M. Rose, "*Post Office Protocol – Version 3",* RFC-1939, May 1996, pp. 1-23, http://www.faqs.org/rfcs/rfc1939.html

Coordination, Cooperation and Conflict Resolution in Multi-Agent Systems

W. Alshabi*, S. Ramaswamy+, M. Itmi*, H. Abdulrab*

***Abstract* - In this paper, we present a condensed survey of multi-agent systems, with special emphasis on cooperation coordination, conflict resolution and closely related issues; issues that are critical for the development of large-scale, distributed complex software systems. Then we present three different cooperative MAS architecture types, discuss their drawbacks and propose the need for a service driven framework for the development of cooperative multi-agent systems.**

***Keywords*: Multi-agent systems, cooperative systems, service computing, coordination, conflict resolution.**

I. Introduction

The development of "stand-alone systems" that solve problems with minimal help from the outside environment have traditionally been brittle in nature. Predominantly AI has encountered such brittleness by injecting more knowledge into the system, including common-sense knowledge, to enlarge the system's range of capabilities. However, attempts to develop larger and more complex and intelligent systems have revealed the shortcomings and problems of centralized, single-agent architectures and current agent based development practices. Such strategies are very shortsighted in general and the ability to flexibly team-up and coordinate group activities toward individual and collective goals is a hallmark of natural intelligence [25]. Research in distributed artificial intelligence (DAI) therefore concentrates on understanding the knowledge and reasoning techniques needed for intelligent coordination, and on embodying and evaluating this understanding in computer systems. Multi-agent systems (MAS), may be regarded as a group of entities called agents, interacting with one another to achieve their individual as well as collective goals. Decentralization causes other serious problems, such as conflicts among the agents and their respective goals. This is because the knowledge contained in each agent might be incomplete, and goals of agents might be in conflict. Therefore, conflict resolution is a critical and implicit problem in MAS. Thus, this paper is an attempt to integrate the various issues and flavors of MAS and propose an enhanced MAS framework for MAS that allows large-scale cooperative behaviors. Our recent work on several related issues are reported in [56-62].

In MAS, agent interaction [80] is generally governed by various needs such as cooperation, competition or coexistence [73, 74, 91] in order to jointly carry out a required task or to achieve a particular goal. Agent interaction may be via direct communication, by means of an intermediary agent or indirectly by actions carried out in the environment. This definition indicates that there are three dimensions that characterize an agent: its goals, its capacities to carry out certain tasks and its available resources. Interactions among agents in MAS are justified by their interdependence accordingly along these three dimensions [75]: (i) Goals Compatibility, i.e. the MAS problem is to determine whether or nor the respective goals of the various agents in the system are compatible. (ii) Agent Capacity, i.e., the MAS problem is task accomplishment through agent interaction. (iii) Resource Relationships, i.e., the MAS problem is the identification and resolution of agent conflicts.

This paper is organized as below: Section 2 different aspects of MAS environments with particular focus on conflict resolution. Section 3 deals with architectures and frameworks for establishing agent cooperation. Section 4 introduces and motivates a service based framework to enable agent cooperation & coordination. Section 5 concludes the paper.

II. Multi Agent Systems, Conflict Resolution and Agent Cooperation

MAS may be comprised of homogeneous or heterogeneous agents, MAS is considered as crucial technology for the effective exploitation of the increasing availability of diverse of heterogeneous and distributed on-line information sources. MAS can also be a framework for building large, complex, and robust distributed information processing systems which exploit the efficiencies of organized behaviour. [2, 3, 13, 15, 31, 37]. Teamwork and communication are two important processes within multi-agent systems designed to act in a coherent and coordinated manner. The need for responsive, flexible agents is pervasive in many application domains due to their complex, dynamic, and uncertain nature of the environment. Sensible Agents [13, 15, 31] are MAS systems designed for domains with a high level of dynamism and uncertainty. Autonomous and interactive characteristics of agents do allow widespread applications for agent-based applications [1, 4, 11, 36]. The immediate application of planning and scheduling to real-world problems has been a motivational factor in using MAS as proof-of-concept for application designs [5, 6, 9, 10].

Conflict resolution (CR) includes conflict detection, search for solutions, and communication among agents to reach an

* LITIS Laboratory, INSA – Rouen, 76131 Mt. St. Aignan Cedex, Rouen, France. This work was supported, in part, by INSA-Rouen – for supporting Dr. Ramaswamy as a visiting professor at INSA during June 2006. Email: waledhaz@hotmail.com, {itmi, habib.abdulrab}@insa-rouen.fr

+ Knowledge Enterprises for Scalable Resilient Infrastructures (KESRI), Computer Science Department, Univ. of Arkansas at Little Rock, AR 72204 USA.. This work was supported, in part, by an NSF (#CNS-0619069) and the Graduate Institute of Technology at UALR. Email: srini@acm.org

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 495–500.

agreement with regard to the CR solution to be pursued. Due to the basic characteristics of multi-agent systems, conflict resolution is a common phenomenon in multi-agent systems [35, 37]. Application domains in which multi-agent system technology is appropriate typically have a naturally spatial, functional or temporal decomposition of knowledge and expertise.

A. MAS and Conflict Resolution

Research in conflict resolution of multi-agent systems has been approached from three distinct perspectives: organization autonomy [43, 44, 48, 51, 56, 58, 59], non-cooperative domains [30, 46] and cooperative multi-agent systems [4, 8, 11, 35, 36, 46, 49, 54, 55]. Research in conflict resolution of cooperative multi-agent systems has been approached from three distinct perspectives: distributed decision-making [52], model description [12, 16, 64-65], and applications [7, 14, 17-19, 20-24, 26, 27, 29, 32-34, 38-42, 45, 47, 50, 53, 63, 66-69].

B. Agent Cooperation

Cooperation is a key MAS concept [72, 77, 79, 80]. Durfee and colleagues [78] have proposed four generic goals for agent cooperation: (i) Increase the rate of task completion through parallelism; (ii) Increase the number of concurrent tasks by sharing resources (information, expertise, devices, etc); (iii) Increase the chances for task completion by duplication and possibility using different modes of realization; (iv) Decrease the interferences between tasks by avoiding the negative interactions. However, cooperation in agent-based systems is at best unclear and at worst highly inconsistent [49]. Researchers like Galliers [82, 83] and Conte [76] underlined the importance of adopting a common goal for agent of cooperation which they consider as an essential element of the social activity. We can characterize a MAS system by the type of implemented cooperation which can range from total cooperation to the total antagonism [84]. Completely cooperative agents can change their goals to meet the needs of other agents. Antagonistic agents, on the other hand, will not cooperate and, their respective goals may be blocked.

III. Architectures, Frameworks for Cooperative MAS

Bond [86] describes the existence of two types of MAS architectures: (i) Horizontal: This structure is useful in some contexts, for example, a situation where a group of agents having different (non-overlapping) capabilities and hence can work towards the goal without needing any conflict resolution. Here all the agents are on the same level with equal importance without a Master / slave relationship. (ii) Vertical: In a vertical architecture, the agents are structured in some hierarchical order. Agents at the same sub-level may share the characteristics of a horizontal structure. The 'horizontally structured' MAS model has several issues –a critical issue is that it quickly becomes too complex and unwieldy for practical applications, wherein agents in the MAS may share some common capabilities. Hence most current frameworks have adopted a hierarchical MAS model (vertical) by organizing the agents in some organizational structure.

A. Frameworks for Cooperation in MAS

Here, we compare three widely used models for agent cooperation in MAS: HOPES, HECODES and MAGIC.

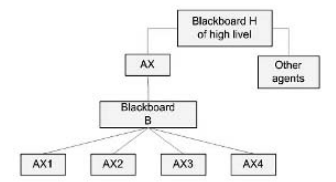


Fig. 1: HOPES Model

HOPES [70] (Hierarchically Organized Parallel Expert System) is well adapted to the needs of problem resolution at a given level. The interaction between agents is carried out by means of blackboard divided between agents at several levels. For example, in Fig.1, agent *AX* will be responsible for the centralization of the results of the cooperation between the agents *AX1, ..., AX4* using blackboard *B*. Results are posted on the blackboard to be used by the agent *AX*, but *B* is also used by the agents of the lower level which, in their turn, interact for the collective resolution of a task. Agents use *B* like a channel during their intra-group communications. Other agents on the same level as *AX* can want to use this result or to collaborate with *AX* in another way. If so, they would use another blackboard *H* of higher level. This stratification can be repeated for blackboards at multiple levels.

HECODES (HEterogeneous COoperating Distributed Expert System) [70] is a suitable environment for a horizontal, hierarchical and recursive co-operation. It is also a blackboard-based system and presents a system for centralized control. The cooperating agents can be heterogeneous with respect to their control strategy, their methods for knowledge representation and the used programming language. In Fig. 2, the expert system, the control subsystem and the blackboard subsystem represent the agent network interconnected by communication channels. Each expert system can provide solutions and solve local problems in an autonomous way by using its own domain-specific knowledge. However, each one would need the assistance of the other agents or can provide services to other agents through its control subsystem. There are interfaces between the expert system and the control subsystem, which are used as communication interfaces or man-machine interfaces for the management of heterogeneous expert system. The blackboard subsystem centralizes the information shared by the expert systems. The control subsystem is responsible for the management of the cooperation and the communication between the agents and monitoring execution times. In practice, the subsystems of blackboard and control are gathered at a central location to minimize

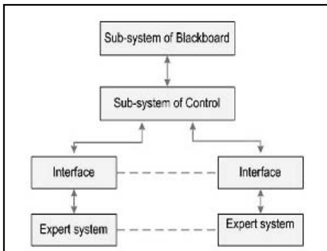


Fig. 2: HECODES

communication delays.

A model of hierarchical multi-agents architecture (MAGIC) is another model of hierarchical agent organization [87-88]. In this structure it is possible to setup powers of delegation between agents, thus facilitating the development. In MAGIC, an agent is an entity having a certain number of competences. These competences make it possible for the agent to hold a role in a MAS application. The competences of an agent can be evolved/moved dynamically (by exchanges between agents) during its existence, which implies that agents can play different roles (thus improving their stature) within the MAS. An agent is dynamically built from an elementary agent "vacuum", by enriching itself by competency acquisition. MAGIC uses an agent termed supervisor that is charged with connecting various qualified agents and in allowing an invocation of method (for competence). The invocation of competences implies that an agent has to achieve a task which requires the exploitation of a certain number of competences that it may (or may not) have by itself. This means that the mode of invocation of competences is established by a hierarchical organization. As a consequence, when an agent seeks to realize a competence it does not necessarily have to know explicitly the agent that will carry out a search for this competence. The supervisor is given the responsibility to find the competence necessary to address this need. The invocations are not named and one agent will not have to know "who makes what". The principle interaction mechanism is therefore, as follows:

1. If the agent has the required competence, it calls upon this directly. (i) If the existence of the competency at another agent is known, an appropriate support request is initiated. (ii) Otherwise, help of the supervisor is sought. If the supervisor knows of the existence of the competency at another agent, an appropriate support request is initiated. (iii) If not, the superior recursively tries to find another qualified supervisory member from higher up in the hierarchy, and then the same mechanism is reapplied.

In MAGIC, the reorganization of the agents' structure can be to some extent dynamic. Bonds between agents can be created when relations appear between two agents of the hierarchy (Fig. 3). That causes to remove the recurring communications along the tree structure and this allows direct agent communication. The decision of creating such links is a prerogative of the agents themselves.

B. Shortcoming of Above Frameworks

HOPES and HECODES are based on a blackboarding technique [89]. The agent is reduced to a knowledge source and the system is made up of the entire rules that form the base of knowledge. Blackboard allows a centralized control and hence can be more effective. But on the other hand, it represents a classic bottleneck and the sources of knowledge are not locally available. Agents do not have any mechanism for direct communication for information exchange and to cooperate transparently. The centralization by means of a blackboard allows for indirect communication via a shared structure and hence has a rigid centralized structure. MAGIC is more appropriate for better resolution of distributed problems with autonomous agents. Nevertheless, MAGIC suffers from several weaknesses for real world applications. (i) *Acquaintances link*: Specific links are established between agents, they allow for direct communications. This means that if new resources are added, or if other agents which are better suited to a need arrive within a group, this may be unknown to the agent. This may cause performance degradation over time. (ii) *Zone Restriction:* When there is a task competency required that an agent does not possess, we need to traverse the hierarchy in order to seek the necessary competence. MAGIC places no bounds on the search scope. This can be a time-consuming activity and not very appropriate for applications that may require 'time-sensitive' answers. Selectively exploring availability of necessary competencies in restricted 'search zones' is impossible. In the worst case, if the required competency is non-existent, the request can reach a standstill at the root until a qualified agent joins the structure. (iii) *Redundant Competencies:* If there are several agents in the system that possess the same competences, there is no mechanism that allows for choice of the agent that will carry out the needed task. The first agent that is identified is the one that is used. In most practical applications, it is imperative to know which agent will carry out the task effectively and a choice/selection mechanism for using quite precise comparative criteria needs to exist. (iv) *Essentiality of a Supervisor:* Similar to a blackboarding system, a supervisor makes it possible to coordinate agents' interactions. However there is no mechanism to detect the breakdown of a supervisor agent; hence some agents can become isolated.

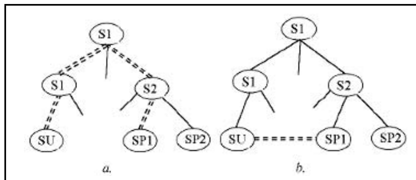


Fig. 3. MAGIC

IV. Our model

To generate a coherent total behaviour of the MAS, the cooperation between agents requires an elaboration mechanism of coordination and so avoids potential conflicts and supports the synergy of the agents' activities while enabling them to profit from their respective capacities, and to benefit from the actions of the ones and others.

For this reason, we introduce the concept of *coordinator agent* into our model. It plays a central function in our approach. It constantly holds the total state of its group but has a partial and limited vision system. Its role is to manage the interactions of its group of agents (the under-hierarchy) and to ensure their coordination like guiding each sending of request for resource towards the suitable local staff. It makes it possible to gather and put in connection several local staff. Each *coordinator agent* has a table made up of the various local staff of its group indexed by the type of resource that it manages (fig. 4).

We mentioned earlier that Durfee and colleagues [78] have proposed four generic goals to establish agents' cooperation. However, an important issue to consider with respect to supporting tasks (ii-iii) is that agents need to be designed with the inherent ability to 'share' pertinent information with other agents in the environment. The resulting efforts by another agent in establishing successful agent cooperation can be vastly improved. Hence, we propose an alternative model in which each agent can be an autonomous entity having control over its own resources and associated skills which enable the agent to cooperate, to communicate and interact with the other agents. Agents can be divided into many groups, and they can cooperate in different levels. Each group of agents consists of several cooperative agent members and a public superior member; this last detains the global state of its group but possesses a partial and limited vision of the system.

Our model comes over the limitations of hierarchical agents architectures, also allows for intelligent information transfer among multiple agents that provide appropriate services. Identification of such information is the key to orchestrating cooperative interactions between agents. Each individual agent in the community of cooperating agents can then appropriately and contextually define its choice of balance (one of the gray triangles in Fig. 5) between cooperation and autonomy. The two competing goals in MAS namely agent autonomy versus cooperation in MAS are illustrated in Fig. 5. Then each agent is capable of providing services autonomously. An application over such an agent network is then represented by a dynamic tree-based data structure of agents that receive and provide various services to accomplish a task / goal. The roots of the tree are coordinator agents. The agents' connections of bonds, which can be between local agents or coordinators agents, form the group. Intermediate and roots nodes are appropriate coordinator agents, which has a pseudo-global view of the system. A bidirectional link represents the existence of a bidirectional channel of communication between two agents.

A. Advantages of Proposed Framework

Our proposed model vastly differs from the HOPES and HECODES models in that all communication between the agents of the group are directed by dynamic coordinator agent and not by a static blackboard, which completely eliminates the disadvantages of the blackboard. Coordinator agents have a principal role: requests routed through these agents offer the best possible solution. Nevertheless, since the links are created at run time, a bottle-neck at the coordinator agent should be avoided. An agent will be considered as 'unavailable' if it does not answer a request over a certain time. At the time of research of a solution, the request can contain a delimitation of the research zone. This is an important point in the case of transport. In case, we have several agents that have the same types of resources, a mechanism of decision based on the negotiation is used to define the optimal solution. In order to have an indirect communication, agents can be divided in several groups, that aims to reduce the number of communication and to have a less complex structure.

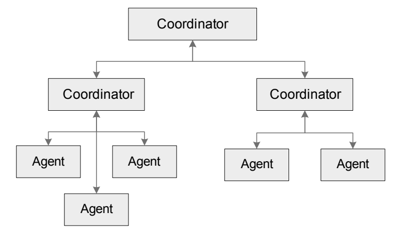


Fig. 4. The dynamic hierarchical model

They allow for a coherent coordinated behaviour of the MAS. Since the interactions are at run time, there needs to be appropriate load balancing mechanisms available. Agents may be able to self-augment a request with appropriate work zone / time delimitations. When several agents with similar resources can provide a needed service, a costing mechanism can be used. The self-aggregation of agents in several groups is nicely supported, communications are also less complex.

In case of conflict, the agent must enter into a negotiation with the conflict group. Various protocols for negotiation exist in the MAS literature. In [59], three different techniques for complex negotiations are presented. These include: (i) negotiation through an arbitrary leader - in this method an arbitrary leader is selected for arbitrating the conflict (interference) resolution process between the agents, (ii) negotiation through chaining – in this method a ranked order assigned to each agent based on when they join the group is used for conflict resolution, and (iii) negotiation through cloning - in this method each agent creates a 'restricted' clone (agent for negotiation) and passes them to every other agent in the group.

V. Conclusions

Cooperation is a key process for multi-agent systems research. In this paper, we have presented a hierarchical multi-agents model which aims to overcome the limits of classic multi-agent architectures. It allows intelligent information transfer among multiple agents and to answering for other applications which were not concerned by the previous models.

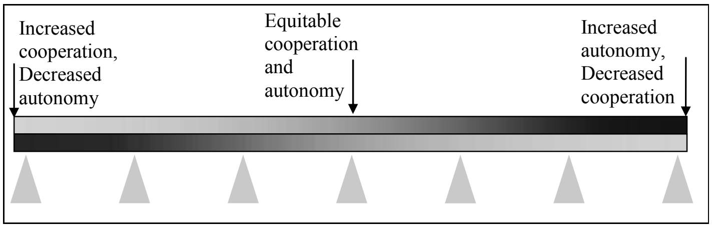


Fig. 5. Agent Cooperation versus Autonomy

References

[1] S. I Morisbak and B. Tessem, "Agents For Case-Based Software Reuse," Applied Artificial Intel., 15(3): 297-332, 2001.

[2] K. E. Biggers and T. R. Ioerger, "Automatic Generation of Communication and Teamwork within Multi-Agent Teams," Applied Artificial Intelligence, 15(10): 875-916, 2001.

[3] G. Nitschke, "Cooperating Air Traffic Control Agents," Applied Artificial Intelligence, 15(2): 209-235, 2001.

[4] P. Stone, M. L. Littman, S. Singh, M. Kearns, "ATTac-2000: An Adaptive Autonomous Bidding Agent," Jour. of AI Res., 15: 189-206, 2001.

[5] Garrido and F. Barber, "Integrating Planning and Scheduling," Applied Artificial Intelligence, 15(3): 471-491, 2001.

[6] P. Priore, D. De La Fuente, R. Pino and J. Puente, "Learning-Based Scheduling Of Flexible Manufacturing Systems Using Case-Based Reasoning," Applied Artificial Intelligence, 15(10): 949-963, 2001

[7] K. S. Barber, T. H. Liu, and S. Ramaswamy, "Conflict Detection During Plan Integration for Multi-Agent Systems," IEEE Trans. on Systems, Man, and Cybernetics, 31(4): 616-628, August 2001

[8] R. Liu, M. J. Shih, and Y. Kao, "Adaptive Exception Monitoring Agents for Management By Exceptions," Applied Artificial Intelligence, 15(4): 397-418, 2001.

[9] J.L. Ambite, and C. A. Knoblock, "Planning by Rewriting," Jour. of Art. Intel. Res., 15: 207-261, July-Dec. 2001.

[10] I. Refanidis and I. Vlahavas, "The GRT Planning System: Backward Heuristic Construction in Forward State-Space Planning," Jour. of Artif. Intel. Res., 15: 115-161, July-Dec. 2001.

[11] B. Zhang and Y. Seo, "Personalized Web-Document Filtering Using Reinforcement Learning," Appl. Artif. Intel., 15(7): 665-685, 2001.

[12] J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. V. Torre, "The BOID Architecture: Conflicts Between Beliefs, Obligations, Intentions and Desires," Proc. of the 5th Intnl. Conf. on Autonomous Agents 2001, pp. 9-16.

[13] K. S. Barber, R. McKay, M. MacMahon, C. E. Martin, D. N. Lam, A. Goel, D. C. Han, and J. Kim, "Sensible Agents: An Implemented Multi-Agent System and Testbed," Proceedings of the Fifth Inter. Conf. on Autonomous Agents 2001, pp. 92-99

[14] K. S. Barber, T. H. Liu, A. Goel, and C. E. Martin, "Conflict Representation and Classification in a Domain-Independent Conflict Management Framework," Proc. of the Third Inter.Conference on Autonomous Agents 1999, pp. 346-347.

[15] K. S. Barber, J. Kim, "Constructing and Dynamically Maintaining Perspective-based Agent Models in a Multi-Agent Environment," Proc. of the 3rd Intnl Conf. on Auto. Agents 1999, pp. 416-417.

[16] H. Jung, M. Tambe, S. Kulkarni, "Argumentation as Distributed Constraint Satisfaction: Applications and Results," Proc. of the Fifth Intnl Conf on Autonomous Agents 2001, pp. 324-331.

[17] C. Huang, J. A. Ceroni, and S. Y. Nof, "Agility of Networked Enterprises - Parallelism, Error Recovery and Conflict Resolution," Computers in Industry, 42(2-3): 275-287, 2000.

[18] Y. Yan, T. Kuphal, and J. Bode, "Application of Multi-agent Systems in Project Management," International Journal of Production Economics, 68(2): 185-197, 2000.

[19] . G. W. Tan, C. C. Hayes, and M. Shaw, "An Intelligent-Agent Framework for Concurrent Product Design and Planning," IEEE Tran. on Engineering Management, 43(3): 297-306, 1996.

[20] E. C. Lupu, and M. Sloman, "Conflicts in Policy-Based Distributed Systems Management," IEEE Transactions on Software Engineering, 25(6): 852-869, November / December 1999.

[21] G. J. Pappas, C. Tomlin and S. Sastry, "Conflict Resolution for Multi-Agent Hybrid Systems," Proceedings of the IEEE Conference on Decision and Control v2 1996, pp. 1184-1189.

[22] J. K. Kuchar and L. C. Yang, "Survey of Conflict Detection and Resolution Modeling Methods," AIAA-97-3732, AIAA Guidance, Navigation, and Control Conf., Aug. 11-13, 1997, pp. 1388-1397.

[23] J. K. Kuchar and L. C. Yang, "A Review of Conflict Detection and Resolution Modeling Methods," IEEE Transactions on Intelligent Transportation Systems, 1(4): 179-189, December 2000.

[24] S. Cooper and A. Taleb-Bendiab, "CONCENSUS: Multi-Party Negotiation Support for Conflict Resolution in Concurrent Engineering Design," Jour. of Intel. Manuf., 9: 155-159, 1998

[25] E. H. Durfee, "The Distributed Artificial Intelligence Melting Pot," IEEE Trans. on SMC, 21(6): 1301-1306, Nov/Dec 1991

[26] M. A. Vilaplana, C. Goodchild, "Application of Distributed Artificial Intelligence in Autonomous Aircraft Operations," Proc. of AIAA/IEEE Digital Avionics Sys. Conf. 2001, pp. 7B31-7B314.

[27] M. Klein, "Supporting Conflict Resolution in Cooperative Design Systems," IEEE Trans. on SMC, 21(6): 1379-1390, Dec.1991.

[28] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," The MIT Press, Cambridge, Massachusetts, 1998.

[29] A. Wallis, Z. Haag and R. Foley, "A Multi-Agent Framework for Distributed Collaborative Design," Proc. of the 7th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE 1998, pp. 282-287.

[30] G. Zlotkin and J. S. Rosenschein, "Cooperation and Conflict Resolution via Negotiation Among Autonomous Agents in Non-cooperative Domains," IEEE Trans. on SMC, 21(6): 1317-1324, Nov./Dec. 1991.

[31] K. S. Barber, R. M. McKay, A. Goel, D. C. Han, J. Kim, T. Liu, and Cheryl E. Martin, "Sensible Agents: The Distributed Architecture and Testbed," IEICE Trans. on Comm., E83-B(5): 951-960, May 2000.

[32] X. Huang and J.Hallam, "Spring-based Negotiation for Conflict Resolution in AGV Scheduling," IEEE International Conf. on SMC, v1 1995, pp. 789-794.

[33] E. Uchibe, T. Kato, M. Asada, K. Hosoda, "Dynamic Task Assignment in a Multi-agent/Multitask Environment based on Module Conflict Resolution," 2001 IEEE Intnl Conf. on Robotics & Auto., May 21-26, 2001, pp. 3987-3992.

[34] H. Yongtong, L. Ping, Y. Yuhong, Z. Danian, M. Changchao, J. Bode, and R. Shouju, "A Multi-agent System For the Support of Concurrent Engineering," IEEE Conf. on SMC, 1996, pp. 959-964.

[35] S. Arai, K. Miyazaki, and S. Kobayashi, "Controlling Multiple Cranes Using Multi-agent Reinforcement Learning: Emerging Coordination among Competitive Agents," IEICE Tran. on Comm., E83-B(5): 1039-1047, May 2000.

[36] H. Venkataramani Johar, "SoftCord: An Intelligent Agent For Coordination In Software Development Projects," Decision Support Systems, 20: 65-81, 1997.

[37] J. K. Lee and M. W. Jeong, "Intelligent Audit Planning System for Multiple Auditors: IAPS," Expert Sys. w/ App., 9(4): 579-589, 1995.

[38] D. C. Brown, B. V. Dunskus, D. L. Grecu and I. Berker, "SINE: Support For Single Function Agents," Proceedings of the 1995 10th Intnl. Conf. on Apps. of AI in Engineering, Computational Mechanics Inc, Billerica, MA, USA., pp. 525-532.

[39] J. Chu-Carroll, "Conflict resolution in collaborative planning dialogs," Int.. J. of Hum. Comp. Stud., 53(6): 969-1015, Dec 2000.

[40] J. Chu-Carroll and S. Carberry, "Communication for Conflict Resolution in Multi-Agent Collaborative Planning," Intnl.Conf. on Multi-Agent Systems (ICMAS-95) 1995, pp. 49-56.

[41] P. McDonnell, and S, Joshi, "Intelligent hierarchy: A framework for distributed shop floor control," Proc. of 4th Ind. Engg. Res. Conf. 1995. IIE, GA, USA., pp. 808-816.

[42] K. Suzanne Barber, David C. Han, and Tse-Hsin Liu, "Strategy Selection-Based Meta-level Reasoning for Multi-agent Problem-Solving," Paolo Ciancarini, Michael Wooldridge (Eds.): Agent-Oriented Software Engineering, First International Workshop, AOSE 2000, June 2000, LNCS 1957, Springer 2001, pp. 269-283.

[43] K. Suzanne Barber, Anuj Goel, and Cheryl E. Martin, "Dynamic Adaptive Autonomy in Multi-agent Systems," Jour. of Experimental and Theoretical AI, 12(2): 129-147, April 2000.

[44] K. Suzanne Barber, Tse-Hsin Liu, and David C. Han, "Agent-Oriented Design," F. J. Garijo, M. Boman (Eds.): Multi-agent System Engineering, 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW '99, Valencia, Spain, LNCS 1647, Springer 1999, pp. 28-40.

[45] K. S. Barber, T. H. Liu, A. Goel, and S. Ramaswamy, "Flexible Reasoning Using Sensible Agent-based Systems: A Case Study in Job Flow Scheduling," Jour. of PPC, 10(7): 606-615, 1999.

[46] William Kilmer, "Reaching Agreement Among Randomly Interacting Agents: Brain Models, Schemas, and Unitary Responses," Journal of Intelligent Systems, 9(3-4): 203-217, 1999.

[47] J. Sun, Y. F. Zhang, and A. Y. C. Nee, "Agent-Based Product Design and Planning for Distributed Concurrent Engineering," Proc. of IEEE Int. Conf. on R&A, v 4 2000, pp. 3101-3106.

[48] K. S. Barber and C. E. Martin, "Dynamic Reorganization of Decision-Making Groups", Proceedings of the Fifth Internl. Conf. on Autonomous Agents 2001, AGENTS'01, May 28-June 1, 2001, Montréal, Quebec, Canada, pp. 513-520.

[49] J. E. Doran, S. Franklin, N. R. Jennings and T. J. Norman, "On Cooperation in Multi-Agent Systems", The Knowledge Engineering Review, 12(3): 309-314, 1997.

[50] . T. H. Liu, C. Chuter, and K. S. Barber, "Virtual Environment Simulation for Visualizing Conflict Resolution Strategies in Multiple Robot Systems," Proceedings of the Fifth IASTED International Conference Robotics and Manufacturing, May 29-31, 1997, Cancún, Mexico, pp. 154-158.

[51] K. Suzanne Barber and Cheryl E. Martin, "Dynamic Adaptive Autonomy in Multi-agent Systems: Representation and Justification," Int. J. of Pat. Recog. and AI, 15(3): 405-433, 2001.

[52] K. S. Barber, D. C, Han, and T. H. Liu, "Coordinating Distributed Design Making Using Reusable Interaction Specifications," C. Zhang and V. – W. Soo (Eds.): PRIMA 2000, LNAI 1881, Springer-Verlag Berlin 2000, pp. 1-15.

[53] L. Lin-sen, Y. Hai-xun, L. Ji-xun, and T. Ming-an, "Intelligent Resolution of Cooperative Conflict," Chinese Jour. of Aeronautics, 13(1): 24-29, February 2000.

[54] M. Adler, E. Durfee, M. Huhns, W. Punch, and E. Simoudis," AAAI Workshop on Cooperation Among Heterogeneous Intelligent Agents", AI Magazine, 13: 39-42, Summer 1992.

[55] V. R. Lesser, "Cooperative Multi-agent Systems: A Personal View of the State of the Art," IEEE Tran. on Know. and Data Engineering, 11(1): 133-142, January/February 1999

[56] S. Ramaswamy , K. Srinivasan , P. K. Rajan , R. MacFadzean , S. Krishnamurthy, "A Distributed Agent-based Simulation Environment for Interference Detection and Resolution", Spl. Issue on Software Agents and Simulation, SIMULATION, June 2001.

[57] R. Donath, A. Dean, D. Girardi and S. Ramaswamy, " Distributed Simulation of an Automated Highway System with Intelligent Vehicles ", Sum. Comp. Sim. Conf., 2001.

[58] S. Krishnamurthy, S. Ramaswamy, P.K. Rajan, " Implementation and Evaluation of a Dynamic Switching Protocol for Interference Resolution in Naval Radar Units", Sum. Comp. Sim. Conf., 2001.

[59] V. Ramachandran, S. Ramaswamy, P.K. Rajan, Complex Negotiation Protocols for a Distributed Simulation Environment, Sum. Comp. Sim. Conf., 2001.

[60] S. Ramaswamy, Yi Yan, "Interactive Modeling and Simulation of Virtual Manufacturing Assemblies: An Agent-Based Approach", Spl. Issue on "Computer Integrated Manufacturing Systems: Recent Developments and Applications", Jour. of Intelligent Manufacturing, Vol. 10, No. 6, December 1999. pp.503-518.

[61] R. Maarfi, A. Dean, E. Brown, S. Ramaswamy, "A Three-tier Communication and Control Structure for the Distributed Simulation of an Automated Highway Systems", 2002 PERMIS Workshop, NIST, MD. August 13 - 15, 2002.

[62] S. Ramaswamy, R. Neelakantan, "Software Design and Testing Using Petri Nets: A Case Study Using a Distributed Simulation Software System", 2002 PERMIS Workshop, NIST, MD, 2002.

[63] R. Thomson, "Deontic Login as Founded on Tense Logic, In. R. Hilpinen, Editor, New Series in Deontic Logic: Norms, Actions and the Foundations of Ethics, D. Riedel, pp. 165-176, 1981

[64] P. Cohen and H. Levesque. Intention is choice with commitment. Artificial Intelligence, 42:213–261, 1990

[65] A. Rao and M. Georgeff. Modeling rational agents within a BDI architecture. In R. Fikes and E. Sandewall, editors, Proc. of the 2nd Conf. on Know. Rep. and Reas., 473–484. Morgan Kaufman, 1991.

[66] J. B. H. Kwa, "Tolerant Planning and Negotiation in Multiagent Environments", Applied Artificial Intelligence, No. 2, Hemisphere Publishing Corporation, 1988

[67] J. B. H. Kwa, "Planning Automated Guided Vehicle Movements in a Factory", PhD Thesis, Univ. of Ediburgh, 1988

[68] E. Brunswik, "Perception and the representative design of psychological experiments", (2nd ed.). Univ. of CA Press. 1956.

[69] C. Sherif, M. Sherif, & R. Nebergall, "Attitude and attitude change: The social judgment-involvement approach", Philadelphia: Saunders. 1965.

[70] D. Bell and J. Grimson, "Distributed Database Systems", Addison-Wesley, 1992.

[71] A.H. Bond, "Distributed Decision Making in Organisation", IEEE Systems, Man and Cybernetics Conference, Nov. 1990.

[72] S. Cammarata, D. McArhur, and R.Steeb, "Strategies of Cooperation in Distributed Problem Solving", In Karlsruhe, editor, Proc. of the 8th Intnl Joint Conf. on AI, v 2, pp 767-770, Aug.1983.

[73] B. Chaib-draa, "Distributed Artificial Intelligence: An overview", A. Ken, J. G. Williams, C. M. Hall, and R.Kent, eds, Encyclopedia Of Comp. Sc. & Tech., v. 31, 215-243. Marcel Dekker, Inc, 1994.

[74] B. Chaib-draa, "Industrial applications of distributed AI", Communications of the ACM, 38(11):49-53, 1995.

[75] B. Chaib-draa, B. Moulin, I. Jarras, "Agent et Systèmes Multiagents", In Principles et architecture des systèmes multi-agents, JP. Briot et Y Demazeau (eds) (Hermes, Lavoisier), 2001.

[76] R. Conte, M. Miceli, and C. Castelfranchi. Limits and levels of cooperation. In Y. Demazeau and J.-P. Müller, editors, Decentralized AI 2 – Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90), pages 147-160. Elsevier Science Publishers B.V.: Amsterdam, The Netherlands, 1991.

[77] Y. Demazeau and J.-P. Müller, editors. Decentralized AI 2 - Proceedings of the Second European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-90). Elsevier Science Publishers, 1991.

[78] E. H. Durfee and V. Lesser. Negotiating task decomposition and allocation using partial global planning. In L. Gasser and M. Huhns, editors, Distributed Artificial Intelligence Volume II, pages 229-244. Morgan Kaufmann: San Mateo, CA, 1989.

[79] E. H. Durfee, V. R. Lesser, and D. D. Corkill. Cooperative Distributed Problem Solving. Vol. 4, page 83-147,Addison Wesley,1989.

[80] J. Ferber. Les systèmes multi-agents, vers une intelligence collective. InterEditions, 1995.

[81] K. Fischer, J. P. Müler, and M. Pischel. A Model for Cooperative Transportation Scheduling. In Proceedings of the 1st Internaional Conference on Multiagent Systems (IC-MAS'95), San Francisco, June 1995.

[82] J. R. Galliers. A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict. PhD thesis, Open University, UK, 1988.

[83] GALLIERS J.R. (1991) Modeling autonomous belief revision in dialogue. In Yves Demazeau et Pierre Müller (éd.) Decentralized Artificial Intelligence 2: Proceedings of the Second European Workshop on Autonomous Agents in a Multi-Agents World, Elsevier Science Pub./North Holland.

[84] B. Moulin and B. Chaib-draa, "An overview of distributed artificial intelligence" In G. M. P. O'Hare and N. R. Jennings, editors, Foundations of Distributed AI, pages 3-54. John Wiley & Sons: Chichester, England, 1996.

[85] R. G. Smith, "The contract net protocol: High-level communication and control in a distributed problem solver", IEEE Transactions on Computers, C-29(12):1104-1113, December 1980.

[86] A. H. Bond, "Distributed Decision Making in Organisation" IEEE Transactions on SMC Conference, November 1990.

[87] N. Bensaid and P. Mathieu, "A Framework for Cooperation in Hierarchical Multi-Agent Systems", Mathematical Modeling and Scientific Computing. Vol 8. 1997.

[88] N. Bensaid and P. Mathieu, "An Autonomous Agent System to Simulate a Set of Robots Exploring a Labyrinth", Proceedings of the 11th Intl. FLAIRS Conference, FLAIRS'98. AAAI Press

[89] P. Nii "Blackboard systems", In Handbook of Artificial Intelligence. Volume IV. Addison-Wesley: Reading, MA, 1989.

[90] A.H. Bond, L. GASSER L , "Readings in Distributed Artificial Intelligence", Morgan Kaufmann, San Mateo, CA, 1988.

New Approach to overcome the complexity issues raised by Simple Bitmap Indexing

Yashvardhan Sharma* Navneet Goyal* Som Ranjan Satpathy*

*Department of Computer Science and Information Systems, BITS, Pilani, INDIA 333031
E-mail: {yash, goel,f2003485}@bits-pilani.ac.in

Abstract-**Recently Data Warehouse System is becoming more and more important for decision makers. Most of the queries against a large Data Warehouse are complex and iterative. The ability to answer these queries efficiently is a critical issue in the Data Warehouse environment. If right Index Structures are built on the columns, performance of the queries, especially ad-hoc queries will be greatly enhanced. In this paper, we have concentrated on various implementation issues of Simple Bitmap Indexing and their analysis.**

1. INTRODUCTION

Data Warehouses are large, special purpose databases that contain historical data integrated from a number of independent sources, supporting users who wish to analyze the data for trends and anomalies. The process of analysis is usually done by queries that aggregate, select and group the data in a number of ways. Efficient query processing is critical because the Data Warehouse is very large, queries are often complex and decision support applications typically require interactive response times [2].

This efficient query processing can be done using Bitmap Indexing techniques, which help us improve the time and space complexities of the vast and complex queries that Data Warehouses deal with. This simplification of query processing in turn speeds up the analysis process which is vital to any large organization [4, 5].

Simple Bitmap Indexing is one such technique. It primarily functions by representing the huge amount of information stored in Data Warehouses as Bitmap Vectors which are nothing but binary representations of the data stored and the processing of binary data is many folds faster than the processing of textual data. But even this process has a bottleneck and in this paper, our focus is on how to deal with this bottleneck by adding more to the concept of Simple Bitmap Indexing which is discussed in detail in the forthcoming sections.

2. PROBLEM STATEMENT

Though Simple Bitmap Indexing simplifies query processing and improves time complexity thereby speeding up the processing of complex real time adhoc queries, it has a major bottleneck of dealing with space efficiently caused by huge cardinality Datasets. In this paper, we are proposing a solution to rectify the flaw and thus improve the space complexity. This leads to better time complexity and faster query processing as well.

2.1 SIMPLE BITMAP INDEX

The Simple Bitmap Index consists of a collection of bitmap vectors each of which is created to represent each distinct value of the indexed column. The i^{th} bit in a bitmap vector, representing value x, is set to 1 if the i^{th} record in the indexed table contains x.

For more detailed definitions:

· A Bitmap for a value: an array of bits where the i^{th} bit is set to 1 if the i^{th} record has the value.

· A Bitmap index consists of one bitmap for each value that an attribute can take [1, 3].

In our discussion, we are considering a simple example of a student database. The table will have student NAME, student AGE, student DISCIPLINE, student CGPA and student CLASS as its columns of which the CLASS and AGE columns will be the indexed columns.

Here we notice that students belong to five different age groups i.e of age 18,19,20,21 and 22. Thus, the attribute AGE has five distinct attribute values. We represent AGE attribute and the corresponding students belonging to particular age by the following simple bitmap:

Example:

18: 0000100100000110000000
19: 1000010000000000100011
20: 0010000100010000000000
21: 0000000011000000011000
22: 0101001000010000010000

We need five bitmaps because we have five distinct attribute values. Each bitmap consists of 22 bit values since our example comprises 22 records for students. For instance, the last bit of the bitmap of age 21 is set to 0 because the last student is not of age 21.

A legal question appears now: how do we retrieve data with the help of a bitmap index? Identify all 1s in the bitmap and find the corresponding record numbers to get the actual records.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 501–503.

This scheme works well till we have less number of distinct values in the columns. As we don't have any limitation on the number of records in the Data Warehouse, the size of simple bitmap index can be very huge if we have large number of distinct values in the columns. So, a space overhead is clearly arising and this can ultimately slow the query processing. Even dealing with binary files can lead to huge space overhead with increasing cardinality of the datasets in a Data Warehouse. This bottleneck can be brought under control only by arranging data and by using efficient data structures to reduce the space requirements of Simple Bitmap Index. Though the cardinality keeps rising still the processing time is kept under control with the space requirements being kept at minimum possible levels. We would discuss how this can be done in detail in the forthcoming section.

3. IMPLEMENTATION ISSUES AND ANALYSIS

After the introduction about Simple Bitmap Indexing technique and its bottleneck, we propose the solution and its implementation and present the results.

To begin with we constructed random data comprising of student information with "Age" and "Class" as the indexed columns. After generating data, we need to look for distinct elements of "Age" and "Class" in order to create bitmap indexes to process queries. The bitmap on each of the columns can be generated by constructing bitmap vector for each of the distinct elements of each indexed column. Indexes are stored in binary files for better storage and processing. Query processing is done by asking the user for the student age and/or class whose information he/she wants to use in his/her decision-making. This input is taken through the console application program. Now, based on the indexed columns and using the AND and OR binary operators according to the given situation, the required information is thrown out for the user. This method of generating bitmaps for all the records and then going for query processing can be more space and time consuming

The time complexity graph for this approach is shown as follows.

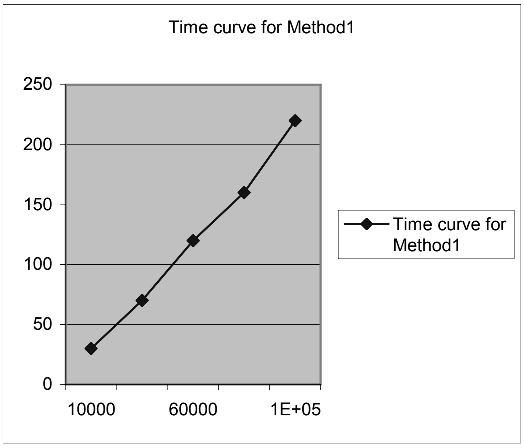


Fig. 1. Method1

Method1: Normal implementation by generation of Simple bitmaps on corresponding indexed columns.

Here the X-axis represents the number of Tuples and the Y-axis represents the query processing time in milliseconds.

From the above approach we can see how the increase in cardinality is drastically affecting the query processing time. For processing a query over 10,000 records is taking around 25 milliseconds while to process a query over 100,000 records is taking around 250 milliseconds. That is about 10 times more and is the direct effect of increasing cardinality.

To solve this particular overhead we can go for the second approach defined below.

Method2: This method represents the improved implementation that saves time by storing only two locations (start and end) of each of the distinct indexed column entries.

This alternative to "**Method1**" can save a lot of space. In this improved method we need to first sort the data based on the distinct elements and then store the beginning and ending locations of each of the distinct elements of a particular record in a data structure and then we can go for query processing. This new method will save a lot of space as we are just storing two locations for each of the distinct elements of the indexed columns and thus improve the query processing time.

The output of this approach on a student database with 80,000 records will be as follows:

For age = 18 start index = 0, end index = 15950
For age = 19 start index = 15951, end index = 31934
For age = 20 start index = 31935, end index = 48061
For age = 21 start index = 48062, end index = 64095
For age = 22 start index = 64096, end index = 79999

For class = 9 start index = 0, end index = 39708
For class = 10 start index = 39709, end index = 79999

Enter your query(attribute1,value,operation,attribute2,value): a 18 & c 9

The above method clearly illustrates the processing done by just storing the first and last indices of each distinct indexed

This page intentionally blank

column (age and class) entries and thus removing the bottleneck.
The time complexity graph for this approach is shown as follows.

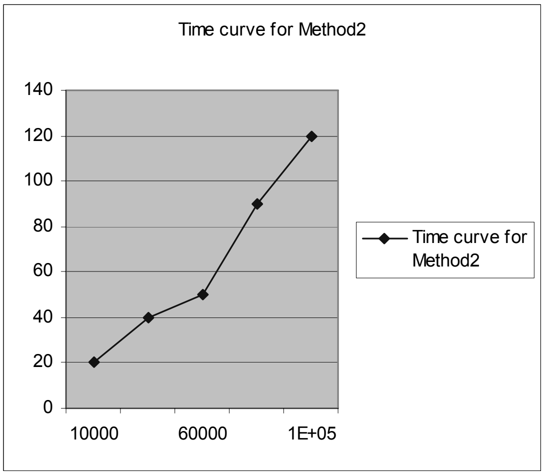


Fig. 2. Method2

Here the X-axis represents the number of Tuples and the Y-axis represents the query processing time in milliseconds.

From the above approach we can see how the increase in cardinality is being dealt efficiently. For processing a query over 10,000 records is taking around 20 milliseconds while to process a query over 100,000 records is taking around 120 milliseconds. That is about 6 times more and is almost half the amount of time taken to process queries with the previous approach (Method1).

The time complexity comparison for both approaches by taking both time curve graphs together:

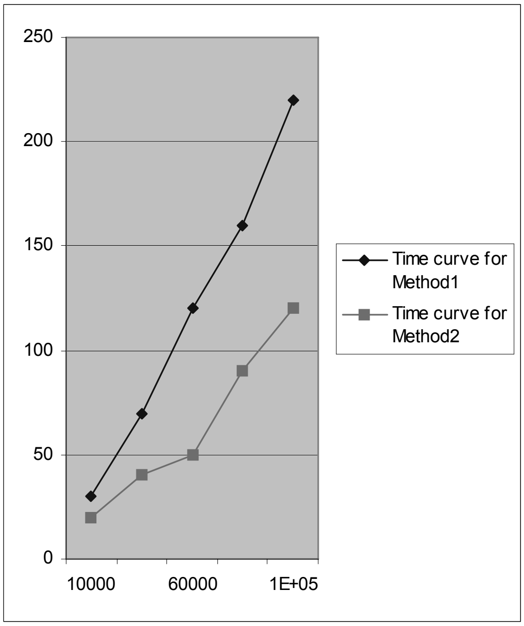


Fig. 3. Comparison

Here the X-axis represents the number of Tuples and the Y-axis represents the query processing time in milliseconds.

4. CONCLUSIONS

To sum up, we can say that there is no basic index that is best suited for all applications. Each application will have its own set of specifications, which we need to follow. The Simple Bitmap Indexing works well with low cardinalities. To deal with high cardinalities we have shown how to overcome the overhead through the new second approach. Compressed Bitmap Indexing is also a promising technique to overcome this problem. Nowadays there are several fast algorithms for evaluating Boolean operators on compressed Bitmaps. Another issue is that we also need some other efficient encoding techniques to lower the number of logical operations.

REFERENCES

[1] C.Y. Chan, and Y.E. Ioannidis, “Bitmap Index Design and Evaluation*”, In Proceedings of ACM SIGMOD Conference*, Seattle, Washington, 1998, pp. 355-366.
[2] 2. P. O’Neil, “Informix and Indexing Support for Data Warehouses.” *Database Programming and Design*, Vol. 10, No. 2, February 1997.
[3] K. Stockinger. Bitmap indices for speeding up high-dimensional data analysis. *In DEXA 2002*. Springer-Verlag, 2002.
[4] Kesheng Wu, Wendy Koegler, Jacqueline Chen, and Arie Shoshani. “Using bitmap index for interactive exploration of large datasets”. *In Proceedings of SSDBM 2003*, Cambridge, MA, USA, 2003, pp. 65–74.
[5] Ramakrishnan, Gehrke, *Database Management Systems*, Third Edition, McGraw-Hill Press,2004.

A Game Theoretic Analysis on Incentive for Cooperation in a Self-Repairing Network

Yoshiteru Ishida[1, 2]
[1] Department of Knowledge-Based Information Engineering,
Toyohashi University of Technology,
Tempaku, Toyohashi, 441-8580 Japan
[2]Intelligent Sensing System Research Center
Toyohashi University of Technology,
Tempaku, Toyohashi, 441-8580 Japan

***Abstract*-This paper discusses when selfish agents begin to *cooperate* instead of *defect*, taking a specific task of maintenance of themselves. The network cleaning problem where a collection of agents capable of repairing other agents by overwriting its content can clean the collection will be discussed. With this problem, *cooperate* corresponds to repairing other agents and *defect* to not repair. Although *both defect* is a Nash equilibrium: no agent is willing to repair others when only the repair cost is involved in the payoff, agents may cooperate with each other when system reliability is also incorporated in the payoff and with certain conditions satisfied. Incentive for cooperation will be stronger when further system wide criterion such as availability is involved in the payoff.**

I. Introduction

Unexpected growth of the large-scale information systems such as the Internet suggests that an open and evolutionary environment for selfish agents will lead to collective phenomena. Internet is undoubtedly the most complex and large-scale artifact that human has ever invented. Observing how the internet has been built and grown up suggests that systems of this complexity may be built not by a usual design but by its own growth logic that the designer even did not think of before its maturation: a synthetic view that a self-repairing network could be embedded in the field.

After the Internet forms itself as a field that allow many selfish activities, several utilities and protocols converges on what may be called "Nash equilibrium" from where no players want to deviate [1]. In the seminal paper by Papadimitriou [2], a problem for the network protocol is posed, which will lead to economic models that allow TCP as an equilibrium point. These studies shed a new light on computational intelligence. That is, rather than implementing an intelligent program, design a field in the Internet that allows intelligent systems to emerge as Nash equilibrium of the Internet field.

Further, the game theoretic approaches to the Internet reveal that obtaining some Nash equilibrium is computationally difficult. This fact, looked from reverse side, would indicate that a computationally difficult task may be solved by selfish agents. Resource allocation, for example, which is computationally tough, may be solved by a market mechanism where many selfish agents participate in. Mechanism Design, a subfield of economics, has been studied [3] and has been extended to Algorithmic Mechanism Design [4] and to Distributed Algorithmic Mechanism Design recently [5].

This article aims a very first step toward embedding a computational intelligence in the Internet field by selfish agents. That is whether selfish agents can ever cooperate and even converge on some tasks. Selfish routing and task allocation have been studied extensively in computational game community, then can agents ever take care of themselves in the first place? We first pose the problem of self maintenance in an agent population, and then a game theoretic approach will be tested whether or not cooperation would occur or under what condition the cooperation will occur.

While this paper amounts to a microscopic analysis focusing on conditions when two interacting agents have incentive to cooperate (i.e. mutually repair), another paper in this volume [22] amounts to a macroscopic studies on the network with many interacting agents.

Section II discusses motivations and paradigm of the present research. The problem of cleaning a self-repair network will be described. Section III discusses the incentives for selfish agents to cooperate based on system reliability and availability of mutually repairing agents that do not have recognition capability. Section IV discusses when and how the selfish agents will cooperate based on the result of Section III.

II. Economic Theory for the Selfish Agents

Game theoretic approach has demonstrated its power in the field of economics and biology. Internet has already reached to the level of complexity comparable to economic system and biological system. Moreover agents approach permits a structural similarity where selfish individual (in economic system of free market) and selfish genes (in biological systems) cooperate or defect in an open network where many things have been left undetermined before the convergence.

Economic approach has been actively studied in Distributed Artificial Intelligence community (e.g. [6, 7]), and application

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 505–510.

to auction may be one of successful domain (e.g. [8]). Economic approach, a game theoretic approach in particular, has been extensively studied in the algorithm and computation community and giving an impact on the network application. Rigorous arguments with the equilibrium concepts, Nash equilibrium among others, are framing a ground theory for economic aspect on the Internet. The cost of the selfish routing has been estimated with how bad the selfish routing might end up to the equilibrium (Nash equilibrium from where no one wants to deviates) relative to the optimal solution. Protocols such as TCP [10], Aloha, CDMA and CSMA/CA have been studied. Packet forwarding strategies in wireless Ad Hoc Networks can also be recast in the framework. Network intrusion detection has also been investigated [11] in a framework of two players' game: Intruder and Defender.

What has been computed by a market mechanism or more generally by collection of selfish agents turned out to be difficult when tried to obtain by computation (as a typical example: Prices of commodities as an index for resource allocations). This fact indicated that the market economy, or more generally free and hence selfish agents properly networked has a potential for computing something that could be difficult when approached otherwise. Also, the fact that the eradication of planned economy by the market economy and that the market economy remains in spite of perturbations indicated that the market economy may be "evolutionarily stable" [12] within these economic systems.

This fact further indicate that a problem solving framework by properly networked selfish agents may have some advantage over other usual problem solving frameworks such as the one organized with a central authority. Also, solutions can be obtained almost free or as a byproduct of the problem solving mechanism, or solutions are almost inseparably embedded in the solving mechanism. The above two observations encourage to recast the problems which have been known to be computationally difficult or the problems difficult to even properly define and approach, such as attaining self-repair systems.

Studies with agents usually assume that agents can be autonomous, hence allowing different rule of interactions: heterogeneous agents. We further assume that agents are selfish in the sense that it will try to maximize the payoff for an agent itself. Thus, agents can be broader than the program or software and they involve users that committed to the agents. Agents may include not only programs but humans (end-point users and providers running Autonomous Systems for the Internet) behind the programs. Mutually supporting collectives may emerge as interplay among agents. Spam mails, computer viruses and worms may be called as (malicious) agents, but they are not mutually supporting collectives. They are rather *parasitic* lone wolves. DDOS attacks and some distributed viruses and worms, however, can be a beginning of the collectives.

The idea developed here can apply not only to the Internet but also to other information network such as sensor networks, as long as they are put in the model.

The models presented in this paper have the following component:

M1. States: Agents have two states (0 for normal; 1 for abnormal). The state will be determined by the action and state of interacting agents.

M2. Actions: Agents have two actions (C for cooperation; D for defection).

M3. Network: Agents are connected by a network and agents can act only on the connected agents (neighbor agents).

Actions may be controlled uniformly or may be determined by the acting agent itself in a selfish agent framework so that payoff assigned to each agent will be maximized. Network may be defined explicitly with a graph or implicitly by specifying the neighbor agents (e.g. lattice structure as in cellular automata and dynamical network as in scale-free networks).

The network cleaning problem considered here assumes the self-repair network composed of nodes capable of repairing other nodes by copying. Since agents do not have recognition capability, source nodes (repairing agents) can be abnormal and target node (agents being repaired) can be normal. Hence mutual repairing without recognition could cause spreading of abnormal states rather than eradication of abnormal states.

Since we focus on the self-maintenance task by mutual repair, cooperation and defection corresponds to repairing and not repairing, respectively.

In the agent based approach, we place the following restrictions similar to immunity-based systems [13]:

– Local information: For each immune cell mounting receptor or a receptor itself (antibody), only matching or not (some quantitative information on degree of matching is allowed) can be provided as information.

– No a priori labeling: For an immune cell or antibody, an antigen is labeled neither as "antigen" nor as "nonself."

Because of these two restrictions, we face the "double edged sword" in this paper, since effectors part (repairing by copying) could harm rather than cure based on local information. This problem of "double edged sword" [14, 15] may be more significant than that of immunity-based systems because we do not assume recognition capability (that could avoid adverse effect) here as in the immunity-based systems. Actions of agents are motivated by selfishness (payoff) rather than the state of the target.

In the following sections, we use a Markov model used for reliability theory as a microscopic model that incorporates M1, M2 and M3 above. The microscopic model focuses on incentive for cooperation retaining network simple with only two agents interaction.

III. A MICROSCOPIC MODEL: NEGOTIATION BETWEEN AGENTS

A. *Prisoner's Dilemma*

In solving the problem of cleaning the contaminated network by mutual copying, another problem (other than "double edged sword") is that each autonomous (and hence selfish) node may not repair others and fall into a deadlock waiting other nodes to repair. The situation is similar to that of Prisoner's Dilemma that has been well studied in the game theory and has been applied to many fields.

The Prisoner's Dilemma (PD) is a game played just once by two agents with two actions (cooperation, C, or defect, D). Each agent receives a payoff *R, T, S, P* (TABLE I) where $T>R>P>S$ and $2R>T+S$.

In Iterated Prisoner's Dilemma (IPD) [16], each iterated action is evaluated in many times. In Spatial Prisoner's Dilemma (SPD) [17, 18], each site in a two-dimensional lattice corresponding to an agent plays PD with the neighbors, and changes its action by the total score it received.

TABLE I
The Payoff Matrix of the Prisoner's Dilemma.
R, S, T, P are payoff to the agent *1*.

agent 1 \ agent 2	C	D
C	*R*(Reward)	*S*(Sucker)
D	*T*(Temptation)	*P*(Punishment)

When the above inequalities are satisfied, the case when both players take action D is a Nash equilibrium from which neither player wants to deviate. In our model, no agent wants to repair other agents. When trapped in this Nash equilibrium, all agents remain silent, and hence all the agents will become abnormal state eventually. And with this state of all agents abnormal, there will be no hope of recovering. Involving system theoretic framework will reveal not only the Nash equilibrium with all agents taking D actions, but the absorbing state with all agents abnormal from which no recovery can happen.

B. *Mutual repairing with selfish agents*

Consider a model with only two agents i (i=1,2) that are capable of repairing other agents. Using conventional notations in reliability theory, λ and μ indicates the failure rate (rate of becoming abnormal) and the repair rate respectively. The *double-edged sword* framework allows agents are capable of repairing other agents, but when the repairing agents are themselves abnormal they will cause the target agents being abnormal (spread contamination) rather than repairing. Thus the state-transition diagram as a Markov model is as shown in Figure 1. Let μ_i denote the repair done by the agent i, and α (<1) indicates the success rate when repair is done by abnormal agent. Repairs by normal agents are assumed to be always successful. The corresponding Kolmogorov equation is:

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{MP}(t)$$

where the time dependent vector variable

$$\mathbf{P}(t) = (p_{00}(t), p_{01}(t), p_{10}(t), p_{11}(t))^T$$

comprised of a component $p_{s_1 s_2}(t)$ denoting a probability of agent *1* being s_1 and agent *2* being s_2 at time t where $s_1, s_2 \in \{0,1\}$ (0: normal; 1: abnormal). $\mathbf{M}$ is a transition matrix corresponding to the state-transition diagram shown Figure 1.

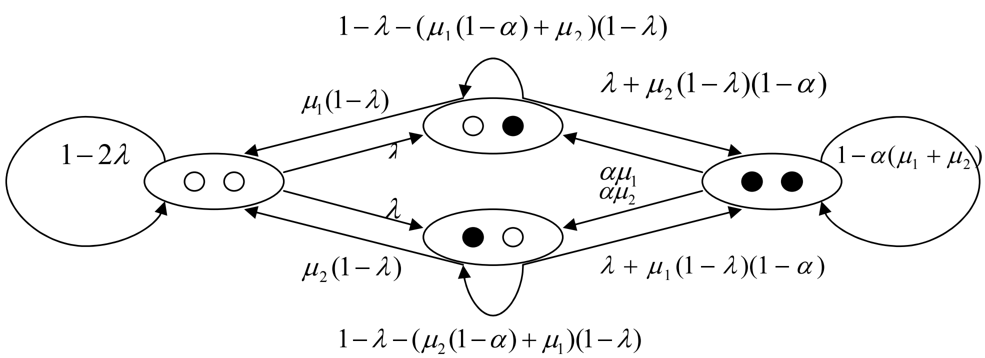


Figure 1. State-transition diagram for the mutual repairing two agents system. White circle indicates normal nodes and the black ones abnormal.

$$\mathbf{M}=\begin{pmatrix} -2\lambda & \mu_1(1-\lambda) & \mu_2(1-\lambda) & 0 \\ \lambda & -\lambda-(\mu_2(1-\alpha)+\mu_1)(1-\lambda) & 0 & \alpha\mu_2 \\ \lambda & 0 & -\lambda-(\mu_1(1-\alpha)+\mu_2)(1-\lambda) & \alpha\mu_1 \\ 0 & \lambda+\mu_2(1-\lambda)(1-\alpha) & \lambda+\mu_1(1-\lambda)(1-\alpha) & -\alpha(\mu_1+\mu_2) \end{pmatrix}$$

For a game theoretic argument, it is further assumed that an agent must decide whether it will repair others or not, corresponding to cooperate and defect in Prisoner's Dilemma. For agent i , $C_i = 1$ if it repair other agent, and 0 otherwise. Let $P_i(C_1, C_2)$ denote a probability of agent i being normal when agent i 's action is C_i. Simple calculation yields the steady-state probability $P_i(C_1, C_2)$ as arranged in TABLE II.

When abnormal agents assumed to do nothing and remain silent as in the case of mechanical systems, then both agent in abnormal state is the absorbing state, and hence the steady-state probability $P_i(C_1, C_2)$ are all 0. In this model, all agents will be abnormal eventually no matter whether cooperation takes place or not. Thus, we assumed even abnormal agents may repair when they take action C.

TABLE II
Steady-state reliability of each agent when mutual repairing is involved. $\beta = \lambda + \mu - \lambda\mu$

	$C_2 = 1$	$C_2 = 0$
$C_1 = 1$	$P_1(1,1) = \dfrac{\beta}{\lambda\left(\lambda + \dfrac{\beta}{\alpha\mu}\right) + \beta}$ $P_2(1,1) = P_1(1,1)$	$P_1(1,0) = 0$ $P_2(1,0) = P_1(0,1)$
$C_1 = 0$	$P_1(0,1) = \dfrac{1}{\lambda + \dfrac{\beta}{\alpha\mu}}$ $P_2(0,1) = P_1(1,0)$	$P_1(0,0) = P_2(0,0) = 0$

TABLE II can be regarded as a payoff matrix of the two players game. If we simply regard $P_i(C_1, C_2)$ as agent i's payoff when actions C_1, C_2 are taken, mutual repairing may happen because of the inequalities:

$$P_1(1,1) > P_1(0,1) > P_1(1,0) = P_1(0,0),$$

$$P_2(1,1) > P_2(1,0) > P_2(0,1) = P_2(0,0).$$

While the action does not make any difference (e.g. for agent *1*, $P_1(1,0) = P_1(0,0)$) when other agent does not cooperate, the agent should certainly cooperate when other agent cooperate (e.g. for agent *1*, $P_1(1,1) > P_1(0,1)$). This is by raising reliability of others, the repairing by them to the self becomes more effective, a circular effect.

Let us take the cost of repairing into consideration. Although both D is Nash equilibrium when there is a positive repair cost, the agents do not have incentive to remain in the both D when the repair cost is negligible.

Let us focus on the payoff. Then agent 1, for example, will choose its action C_1 to maximize:

$$P_1(C_1, C_2) - c \cdot C_1,$$

where c is a cost of repairing relative to the benefit measured by the reliability of itself. Involving cost for cooperation would naturally bias the situation towards more defect-benefiting. When the opponent defects, the agent simply lose the cost of cooperation if it cooperates. However, there is still a chance for mutual cooperation when the opponent cooperates: $P_1(1,1) - c > P_1(0,1)$ holds when the cost relative to benefit satisfies:

$$\frac{(\beta - \lambda)\left(\lambda + \dfrac{\beta}{\alpha\mu}\right) - \beta}{\left(\lambda\left(\lambda + \dfrac{\beta}{\alpha\mu}\right) + \beta\right)\left(\lambda + \dfrac{\beta}{\alpha\mu}\right)} > c.$$

Selfishness of an agent is reflected on the objective function that the agent will maximize, and the reflection is not a trivial task. The above agents are shortsighted in implementing the selfishness. Foresighted agents would consider the event of other agent failure as losing chance of being repaired by the agent, and the extinct of all normal agents as a fatal event that should be avoided by paying high cost. If the repairing by abnormal agents does not happen, extinct of normal agents is an absorbing state from which no other state will arise.

Figure 2 plots the difference $P_1(1,1) - P_1(0,1)$ when the repair success rate by abnormal agent α changes from 0 to 1 and $\lambda = 10^{-4}, \mu = 10^2\lambda$ are fixed. There is a strong incentive for agent *1* to cooperate when the success rate is about 0.1. The incentive decreases almost linearly when the rate exceeds 0.2 in this case, which indicates reliable repairs by abnormal agents promote cooperation.

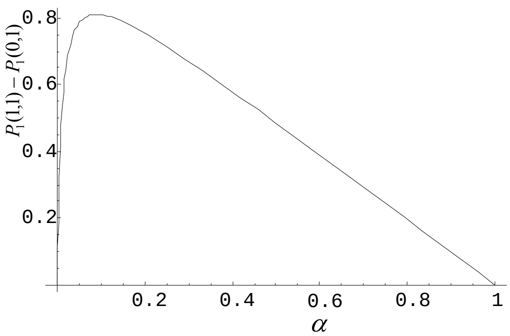


Figure 2. Plot of the difference when the repair success rate by abnormal agents α changes from 0 to 1 and $\lambda = 10^{-4}, \mu = 10^2\lambda$ are fixed.

If the availability (the probability that at least one agent remains normal) is used as a payoff for each agent, then there will be stronger incentive to cooperate when the other agents cooperate, since the difference $AV(1,1) - AV(0,1)$ is larger than the difference $P_1(1,1) - P_1(0,1)$ as shown in TABLE III.

This indicates that even selfish agents will be more likely to cooperate if they take systemic payoff that evaluates cost and benefit in more system wide and longer term; a beginning of self-organization to mutually supporting collectives.

TABLE III

Steady-state availability AV where availability is a probability that at least one agent is normal.

	$C_2 = 1$	$C_2 = 0$
$C_1 = 1$	$AV(1,1) = \dfrac{\beta + \lambda}{\lambda\left(\lambda + \dfrac{\beta}{\alpha\mu}\right) + \beta}$	$AV(1,0) = \dfrac{1}{\lambda + \dfrac{\beta}{\alpha\mu}}$
$C_1 = 0$	$AV(0,1) = \dfrac{1}{\lambda + \dfrac{\beta}{\alpha\mu}}$	$AV(0,0) = 0$

IV. Discussions

The worst case analysis [9] uses Nash equilibrium as a convenient substitute for the solution when tasks are left to selfish agents. The cost for the Nash equilibrium relative to the optimized solution has been proposed to measure the cost of "Anarchy". This paper rather focuses on the self maintenance task, self-repairs by mutual copying in particular, and discusses when selfish agents begin to cooperate. Further discussions are needed on when these selfish agents organize themselves to mutually supporting collectives.

The present research will have two important significances: one engineering and another theoretical. For engineering significance, computing paradigm such as Distributed Computing System [19] Grid computing [20] and Parasitic computing [21] become background. When Grid Computing becomes dominant for large-scale computing, what we call *agents* (autonomous programs that can move from nodes to nodes) will become like *processes* in the Unix OS. One important difference is that agents may be selfish, and will not be organized with a central authority as is done in the conventional OS. Then, the organization of selfish agents will become an organization with an weakest central authority, or even with a distributed authority as seen in the free market economy. Naturally, information processing with selfish agents will be imperative, thus making the game theoretic approach and economic approach such as selfish task allocation and routing important.

Another significance is that it will provide an organizational view for Artificial Life (a life-like form which has some identity hence boundary). Self-organization of selfish agents will be more than a mere collection of independent agents, but rather a cluster of cooperative agents. That would reveal an intrinsic logic and process that selfish agents form multi-agents organisms, similarly to multi-cellular organisms. The game theoretic approach will provide a threshold and a mechanism for defective selfish agents will develop into cooperative selfish agents when payoffs are recast in a broader context of time and space.

V. Conclusions

For large-scale information systems, a game theoretic approach is important, since it will give results concerning what would happen when selfish agents are involved. However, what is "selfish" is open and to be determined in the context and in the environment. This research assumes that selfish agents try to maximize its payoff. Then the next problem is to set the payoff function reflecting the context and the environment. We discussed the cases when only repair cost, system reliability, and more systemic evaluation such as availability are involved. Incentives to cooperation increases when more systemic evaluation is involved in the payoff. Specifically, if agents stick to short sighted payoff such as repair cost, agents will lose the partner that will repair when they become abnormal, or even worse all the agents will be eventually abnormal and they will lose the chance of being repaired forever.

The current research should further developed to studies on how and when the mutually supporting collectives emerge in large-scale information systems such as the Internet.

ACKNOWLEDGMENT

This work was supported in part by Grants-in-Aid for Scientific Research (B) 16300067, 2004. This work was partly supported also by the 21st Century COE Program "Intelligent Human Sensing" of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

REFERENCES

[1] J. Nash. "The bargaining problem," *Econometrica*, 18, pp.155-162, 1950.
[2] C. Papadimitriou, "Algorithms, games, and the Internet," in *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, pp. 749–753. 2001.
[3] N. Nisan and A. Ronen, "Algorithmic mechanism design," *Games and Economic Behavior* 35, pages 166–196 , 2001.
[4] J. Feigenbaum, C. Papadimitriou, and S. Shenker,"Sharing the Cost of Multicast Transmissions," *Journal of Computer and System Sciences* 63, pp. 21–41, 2001.
[5] J. Feigenbaum and S. Shenker, "Distributed Algorithmic Mechanism Design: Recent Results and Future Directions," *Dial-M'02*, September 28, 2002
[6] Y. Shoham and M. Wellman, "Economic Principles of Multi-Agent Systems," *Artificial Intelligence* 94, pp. 1–6, 1997.
[7] W. Walsh and M. Wellman, "A Market Protocol for Decentralized Task Allocation," in *Proceedings of the 3rd International Conference on Multi-Agent Systems*, IEEE Computer Society Press, Los Alamitos, pp. 325–332, 1998.
[8] D. Parkes, "Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency," PhD Thesis, University of Pensylvania, 2001.
[9] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria,".In: *16th Annual Symposium on Theoretical Aspects of Computer Science*, pp. 404-413 , 1999.
[10] A. Akella, S. Seshan, R. Karp, S. Shenker, and C. Papadimitriou, "Selfish Behavior and Stability of the Internet: A GameTheoretic Analysis of TCP," *SIGCOMM'02*, August, 2002
[11] T. V. Lakshman M. Kodialam, "Detecting network intrusions via sampling: A game theoretic approach," In *IEEE INFOCOM*, San Francisco, California, USA, 2003.
[12] J. Maynard Smith, *Evolution and the Theory of Games*, Cambridge Univ. Press, Cambridge, 1982.
[13] Y. Ishida, *Immunity-Based Systems: A Design Perspective*, Springer, Heidelberg, 2004.
[14] Y. Ishida, "A Critical Phenomenon in a Self-Repair Network by Mutual Copying," Knowledge-Based Intelligent Information and Engineering Systems (KES'2005), *Lecture Notes in Artificial Intelligence* (LNAI 3682), pp. 86-92 , 2005.
[15] Y. Ishida and T. Mori, "A Network Self-repair by Spatial Strategies in Spatial Prisoner's Dilemma," Knowledge-Based Intelligent Information and Engineering Systems (KES'2005), *Lecture Notes in Artificial Intelligence* (LNAI 3682), pp. 79-85 , 2005.
[16] R. Axelrod, *The Evolution of Cooperation*, Basic Books, New York , 1984.
[17] M.A. Nowak and R.M., May, "Evolutionary games and spatial chaos," *Nature*, Vol. 359, pp.826-829, 1992.
[18]. Y. Ishida and T. Mori, "Spatial Strategies on a Generalized Spatial Prisoner's Dilemma," *J. of Artificial Life and Robotics*, Volume 9, Number 3, pp. 139 – 143, 2005.
[19] D.J. Farber and K. Larson, "The Architecture of a Distributed Computer System - An Informal Description," Technical Report Number 11, University of California, Irvine, Sept. 1970.
[20] I. Foster, Ian and C. Kesselman. (Eds) The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, Inc., 1999.
[21] A.-L.Barabasi, V.W. Freeh, H. Jeong, and J.B. Brockman, "Parasitic computing," Nature, 412, pp. 894-897, 2000.
[22] M. Oohashi and Y. Ishida, "A Game Theoretic Approach to Regulating Mutual Repairing in a Self-Repairing Network," *International Joint Conferences on Computer, Information and Systems Sciences and Engineering* (CIS2E 06), this volume, 2006.

RESOURCE ALLOCATION AND ITS DISTRIBUTED IMPLEMENTATION

Zsolt Tibor Kosztyán PhD[1], Andrea Bencsik PhD[2], Szabolcs Póta[3]

[1]Assistant Professor, Department of Management, University of Pannonia
[2]Associate Professor, Department of Marketing and Management, István Széchenyi University of Győr
[3] PhD Student, Department of Information Systems, University of Pannonia

ABSTRACT

During the execution of a project (investment, innovation etc.), three important parameters must be kept in mind: we have to execute the project as soon as possible, with minimal total cost and not to exceed resource (manpower, materials, engines etc.) availabilities.

Why does it important to execute the project as soon as possible with minimal total cost? If more than one company compete for the execution of an investment project, usually the chance of winning the tender will be higher if a company can execute the project with minimal total project time (TPT) and minimal total project cost. This problem could already be handled in the 60s and 70s with network planning (CPM, MPM, PERT etc.), scheduling (Gantt Diagrams, LOB etc.) and other related cost-minimizing (CPM/COST, MPM/COST etc.) techniques. The most difficult problem was to handle the resources. During the execution of a project we must keep in view the resources, because these resources are usually straitened. There are well-defined number of labours, engines and so on.

If we would like to execute the project with minimal TPT and minimal total project cost and optimal use of the resources (manpower, materials, engines etc.) the problem becomes easily so hard to solve (already at 5000-10000 activities) that computers available today cannot find the solution within a reasonable time. The real problem is more complicated, because before the execution of the project we can only estimate the duration time, (variable) cost and resource need of activities.

In real life it is common that the duration time of project activities cannot be estimated correctly. In this paper a novel algorithm is introduced by which an optimal resource allocation with minimal total cost for any arbitrary project could be determined. Moreover, this algorithm also handles the competences of the human resources.

A distributed problem solving environment is also introduced that implements the above mentioned optimal resource allocation algorithm with a parallel branch and bound method. The system is built on the Jini technology **[44]**. It is a dynamic, service-oriented infrastructure that utilizes spare cycles of networked workstations in an efficient way and solves computation intensive problems more easily due to the parallelization.

Keywords: Deterministic Resource Allocation, Stochastic Resource Allocation, Distributed Systems, Handling Competences in Resource Allocation.

I. INTRODUCTION

Our novel method schedules the activities in the alternative paths of an optimal resource allocation satisfying a given target function and taking into account that the duration times of the activities are probability variables, which have an expected value and standard deviation **[30]**. According to former studies 10-12% cost can be saved if the duration times of activities are handled as probability variables instead of deterministic values, hence the uncertainty of duration times can be managed and the total project time can approximately be determined if a significance level is given. **[27]** After all, the total project time is often influenced by unanticipated events. In the case when resources and the duration time of activities are changing at projects in progress, a new resource allocation for the running activities and for those still not started can be determined with this method. In this paper we introduce a new algorithm, which refines any feasible solution to determine an optimal resource allocation. This algorithm can be used when the duration times of activities are deterministic or stochastic variables. In the section 3 the deterministic version and section 4 the stochastic version of the resource allocation method will be introduced. In the section 5 a new method will be discussed, which can handle the competences of the human resources.

The main defectiveness of any Project Management software is that they cannot handle the human competences. In real life an activity is prepared by a group, or a subcontractor. What is the optimal size of the groups in terms of resource allocation? How can we collect the adequate people in order to prepare activities earlier? In this paper an algorithm is introduced, which can handle the competences of the human resources.

In the filed of combinatorial optimization or artificial intelligence the method called "branch and bound" is very popular. It is often used in

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 511–518.

such NP-hard optimization problems where the search of the solution with simple enumeration would need long time and large resource capacity that exceeds the capabilities of today's computers. **[6-8]** Typical examples are the knapsack or traveling agent problems, but our resource allocation optimizing problem is of this kind as well. **[4]** One of the advantages of branch and bound (B&B) is that it is only a framework method that defines only the iterative steps and the rules that must be applied at each step. **[10, 11, 41]** Every B&B algorithm can be characterized with four rules: the branching, bounding, problem selection and elimination rules. The rules are only directives and say nothing about the concrete implementation thus the B&B method can be adapted to a variety of problems. We also use this method to find the optimal solution for resource allocation. A more detailed description of the B&B algorithm can be found in **[13, 36, 41]**.

In our system we use a parallel B&B that can significantly decrease the computation time or can achieve a closer solution to the optimal one in the same time when all the solutions are feasible. Moreover, the distribution of the task to different computation sites will result in lower resource (e.g. CPU, memory) consumption at each site, thus can make a problem solvable that was unsolvable on a single machine because of the resource limitations. At a certain class of applications (e.g. at the ones needing many synchronization and inter-process communication) the parallelization does not decrease, rather increase the overall execution time, but the benefit of resource sharing can be more important. The parallelization issues of the B&B algorithm were discussed in many papers **[13, 29]**, the one that we use in our distributed problem solving environment will be discussed in more detail in section 6 from a theoretical approach and in section 7 from the implementation point of view.

II. BACKGROUND

When carrying out a project or a small-scale series production management, we would like to finish a project with minimal total project time and minimal total project cost using resources optimally. For determining the total project time we can use some of the common scheduling methods. Although the scheduling can be easily and quickly solved by a simple computer, the resource allocation problem is much harder and requires more time consuming computation.

For that very reason the heuristic methods are more popular then algorithmic ones. Heuristic methods find a feasible solution. These methods could be much faster, but some times the optimal solution would be important (equalized consuming resources, equalized production etc.). **[2, 3, 6, 8, 22, 32, 42]** Algorithmic methods find an optimal solution, but intermediate steps are usually infeasible solutions. If the computational demand is too high we cannot stop the algorithm in order to get the current feasible solution, because the intermediate step is not surely feasible. **[4, 21, 30]** The evolution methods are starting from a feasible solution and refine the solution, but the optimality is usually not guaranteed. **[27]** The introduced RALL-OPT method also starts from a feasible solution and refines the solution in every step. But this method guarantees the optimal solution in finite steps and the intermediate steps are all feasible. In the next section this new algorithm will be described in detail.

III. DETERMINISTIC RESOURCE ALLOCATION

As it was mentioned earlier our novel resource allocation algorithm solves the given problem with certain resource constraints starting from a feasible solution and continuing until the optimal resource allocation is found. This method primarily handles renewed resources (e.g. human resources). Moreover, considering that the total project time is often influenced by unanticipated events a new resource allocation for the running activities and for those still not started can also be determined with this method. The algorithm can also be used when the availability of resources are not a constant function, or to determine a resource allocation with minimal total project time (TPT) and minimal total project cost, or is capable of using different resources and can be applied in parallel projects

This deterministic resource allocation method schedules the activities in the alternative paths of an optimal resource allocation, satisfying a given target function. In this method the existence of a feasible solution is assumed. The algorithm finds the optimal solution in finite steps, according to the given target function.

3.1 Problem Definition and Notations

Before determining the optimal resource allocation, a target function has to be specified. This function could be the earliest start time of the activities, or the smoothness of the level of required resources and so on. In this paper the activities will be scheduled as early as possible.

$$\min_{\forall(i,j)\in P} x_{(i,j)}$$

subject to $x_{(i,j)} \leq w_{(i,j)}-z_{(i,j)}$, where $x_{(i,j)}, w_{(i,j)}, z_{(i,j)} \in \boldsymbol{R}_o^+$,

$\phi(z_{(i,j)}+x_{(i,j)}) \leq c$, where $c \in \boldsymbol{R}_o^+$, $\phi \in \boldsymbol{R}_o^+ \rightarrow \{r_1, r_2, .., r_n\}$, $r_1, r_2, .., r_n \in \boldsymbol{R}_o^+$, $n \in \boldsymbol{Z}^+$

$x_{(i,j)}$ is the used slack time of the $(i,j)\in\boldsymbol{P}$ activity (it is the difference between the start time determined by a feasible solution and the earliest start time of the activity). $\boldsymbol{P}$ is the set of optimizable activities, $w_{(i,j)}$ is the upper bound of the start time of the activity, $EST_{(i,j)} \leq z_{(i,j)}$ is the lower bound of the activity. □ is the total amount of resources in a given time, c is the resource constraint. The lower and the upper bound can be determined Brucker [5] and Heilmann [23] algorithms.

3.1.1 Searching an Optimal Resource Allocation from a Feasible Solution

Finding an optimal resource constrained resource allocation is an NP-hard problem. Usually a heuristic method finds a feasible solution in a very short time. **[20, 27]**

An optimal resource allocation could be determined from a feasible solution if a target function is given. For instance if the tasks should start as early as possible the target function could be determined and the problem can be solved with a branch and bound algorithm.

In the following chapter this method will be expanded when the constraint of resources is not a constant function. A typical example for this when employees work on the weekends, and that time usually we have fewer amounts of human resources than on weekdays.

3.4 How to Use the RALL-OPT Method

The first step of our method is to determine the logical relationship among the activities. After that we have to estimate the duration time of activities, and then we can use any scheduling algorithms (CPM, MPM etc). In the scheduling phase the lower and the upper bound of start time of activities can be estimated, (and can be further refined in the phase of feasible resource allocation search). Then, if we would like to handle the resources, we have to estimate the demands of resources of activities, and have to determine the resource availabilities for different resources. We can find a feasible solution by any heuristic, or evaluation method. We can refine the lower bound of start time of activities with Brucker's method **[5]**. If the total project time unchanged, after finding feasible solution the upper bound of start time of activities could be the latest start time of the activities. If the start time of any activity is greater than the latest start time then the upper bound will be the actual start time. We can determine the upper bound more accurately with Heilmann's method **[23]**. If a feasible solution exists and we would like to determine an optimal solution, we have to determine a target function (starting time of activities as early as possible). After that we can use our method, which determines the optimal solution. This method uses a parallel branch and bound technique.

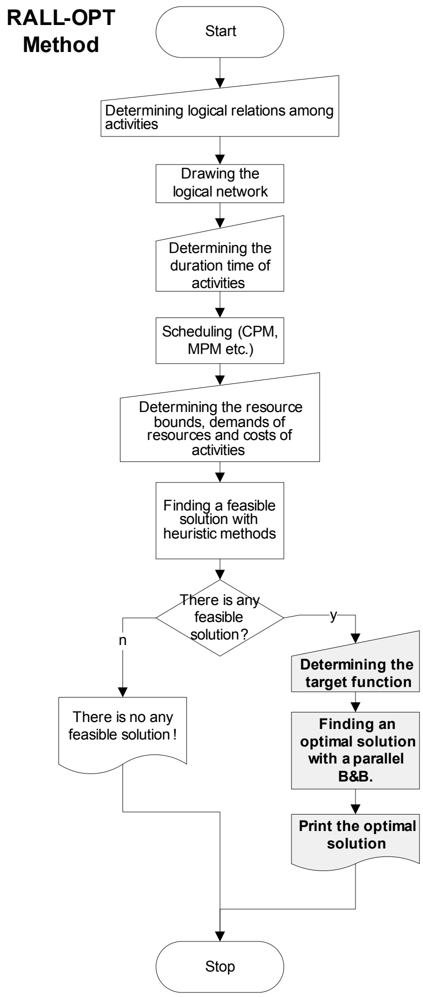


Fig. 1. RALL-OPT method

In the next section we describe the handling of the total cost of the project or production. With this method an optimal total cost of the project can be determined considering the offered remuneration.

IV. STOCHASTIC RESOURCE ALLOCATION

In this chapter a model is introduced which can uniformly handle the uncertainty of the parameters of resource allocation. It is shown that the terms of uncertainty in measurement processes are applicable in project management, too.

4.1 Stochastic time scheduling

In stochastic time management the duration time of activities are handled as stochastic probability variables. Two cases are feasible:

– the distribution of the duration time of activities is approximately known, or
– the distribution of the duration time of activities is completely unknown or at most very slightly known.

If the distribution of the duration time of activities is known, the expected value, variance and standard deviation (standard uncertainty) of the duration times are to be determined as follows **[27]**:

$$E(t_{i,j}) = \bar{t}_{i,j}, D^2(t_{i,j}) = \sigma_{i,j}^2 = u^2(t_{i,j}) \qquad (1)$$

If the distribution of the duration times is unknown, then in the project planning the PERT method can be used and in order to facilitate the calculations it can be assumed that the distribution of the duration times follow a β-distribution.

In the small-scale series production the preparation of a new product can be regarded as a project. In this case there is no information about the distribution of the duration times and many times it cannot be assumed that the distribution of duration times follows a β-distribution. In most cases it is difficult to estimate or determine the expected value, variance and standard deviation (standard uncertainty) of the duration times. The fact that a product is manufactured several times can be of assistance. In this case the expected value can be estimated as follows:

$$E(X_i) = \overline{X}_i = x_i = \frac{1}{N}\sum_{j=1}^{N} X_{i,j}\ , i=1,2,..,M \qquad (2)$$

where M is the number of activities, N is the number of production series and $X_{i,j}$ is the activity i in production j.

The variance of duration time of the activity i is:

$$\mathrm{u}^2(\mathrm{x}_\mathrm{i}) = \mathrm{s}^2(\mathrm{x}_\mathrm{i}) = \mathrm{s}^2(\overline{\mathrm{X}}_\mathrm{i}) = \frac{\mathrm{s}^2(\mathrm{X}_\mathrm{i})}{\mathrm{N}} \qquad (3)$$

Note: The variable cost and demand on resources of activities can be estimated similarly.

After having determined the expected value and standard deviation (standard uncertainty) of the duration times (possibly the variable cost and demand on resources of the activities), then the expected value and standard deviation (combined standard uncertainty) of the total project time can also be determined, provided that these probability variables follow the same distribution. **[1, 12, 15-17, 25, 27, 28, 31, 34, 35, 45]**

4.3 Handling the uncertainty of total project time

In this chapter it is shown how to plan the project if one has to handle the uncertainty of the duration times.

Step 1: Change the expected value of the duration times to make it of minimal variable costs. (If there is a deterministic or stochastic function between the duration times of activities and the required variable costs than the "normal" duration time of the activities with minimal required variable costs can be determined with a simple minimizing method assumed that the duration time - cost functions are convexes.)

Step 2: Draw the PERT-chart and determine the expected value, variance and standard deviation of duration times and the total project time.

Step 3: Decrease the expected value of duration times of activities in the critical paths. Determine the total project times with minimal total cost and duration times.

Step 4: Determine the diagram of resources. Find a feasible and then an optimal solution of the resource allocation problem. (The author described some methods to find optimal resource allocation for different cases earlier. For instance on-line resource allocation, optimal cost resource allocation etc. **[30, 36]**. By these methods optimal resource allocation from feasible solutions can be found when a target function is given.)

(Step 5:) The techniques of network planning and resource allocation are also usable after the planning at the implementation and control. If duration times or demands on resources change, the new optimal resource allocation can be determined with the on-line resource allocation. During the production the expected values of activities have to be replaced by the real values of duration times and value 0 has to be chosen for the standard deviation. This time the uncertainty of the total project time will be decreased. At the end of the program the stochastic network will become deterministic.

In the previous sections in the resource allocation we did not take into consideration that sometimes we have to allocate human resources too. In a project or production there are many employees with different skills, different competences. We can assume that the duration time of activities is influenced by the skills and competences of workers. In the following chapter a new method will be introduced which can handle the competence of human resources.

V. ADAPTING BRANCH AND BOUND FOR THE RALL-OPT METHOD

In the RALL-OPT method finding the optimal solution by a given target function is a crucial point. At projects with hundreds of activities this search would need long time and what is more serious large computational capacity. Clearly this problem is NP-hard and thus the brute force search of the optimal solution by simply enumerating all possible solutions is not feasible by large problems. One most commonly used method is the branch and bound (B&B) method. With this method we can prune large sections of the search-tree by comparing the solution of the current node to the computed upper or lower bound and thus the global optimum can be found in a shorter time as if we would look through the whole tree. One of the advantages of branch and bound (B&B) is that it is only a framework method that defines only the iterative steps and the rules that must be applied at each step, thus it is quite straightforward to adapt it to a variety of optimization problems.

The first step is to define the problem states that will form the nodes of the search tree. Each problem state contains the whole problem description thus the solution of an arbitrary problem state is a feasible solution of the initial problem. By the RALL-OPT method a problem state p_i contains the descriptions of all the activities and some other values that are required for moving activities in the resource chart, the form of p_i is the following:

$$p_i = \left\{ A, P_i, T_i, T_{Si}, c, Q_i, U_i \right\} \quad (5)$$

Where

A: is the base set of activities,

P_i: the set of all optimizable activities; if $P_i = \emptyset$ then a leaf is reached, no more branching can be made

T_i: set of breakpoints in the resource chart with current resource allocation

T_{Si}: previous-activity relations of activities in P_i

c: global resource bound

Q_i: set of activities that are to be optimized in the next step, $Q_i \in \wp(P_i)$

U_i: elements of P_0 that cannot be moved any more because of some bounding conditions, i.e. $U_i = P_0 \setminus P_i$

In the current implementation we have chosen to minimize the overall project duration time which is equivalent with the minimizing of the sum of activity slack times. Thus the solution f_i of a given problem state p_i can be simply formed as the following:

$$f_i = \sum_{P_0} x_{(i,j)} \qquad (4)$$

Where $x_{(i,j)}$ is the slack time of activity (i,j) that is element of P_0, the set of all initially optimizable activities (e.g. the activities on the critical path are not element of P_0). The lower bound g_i of a problem state is computed in a similar way but operating on the current set of optimizable activities:

$$g_i = \sum_{U_i} x_{(k,l)} \qquad (5)$$

Here U_i is the set of activities in P_i that were optimizable initially but reached some bounds and cannot be moved in the further steps. For these activities the slack time is fixed and cannot be decreased any more so the lower bound is the sum of these slack times.

Now we have all the main elements for the branch and bound algorithm, namely the problem state definition (p), the target function (f) and the lower bound function (g). For the complete algorithm, however, three more values must be defined: H is the set of active problem states that are waiting for expansion, p^* is the incumbent, which is the problem state with the best solution so far, that is noted with f^*. With all these notions the branch and bound algorithm that is adapted to our optimal resource allocation problem is the following:

1. We put the initial problem state p_0 to H and compute $f_0 = \sum_{P_0} x_{(k,l)}$.
2. Based on the **selection rule** we choose an active problem state p_i from H.
3. If $Q_i \neq \emptyset$, *where* $Q_i \in p_i$, than
   a. we move the activities in Q_i to as early as possible constrained by the parameters T_{Si} and c of p_i.
   b. count $f_i = \sum_{P_0} x_{(k,l)}$ for p_i
   c. update f^* and p^* if f_i is a better solution

   Note: The condition of step three is not satisfied only at the initial problem state, since $Q_0 = \emptyset$, otherwise $Q_i \neq \emptyset$.
4. Upon the **branching rule** create new sets $Q_{1 \ldots} Q_n$ from P_i, so that $Q_j \subset \wp(P_i)$, $j = 1,\ldots n$ and n is the number of child nodes. If there is such an activity that cannot be scheduled into earlier time in the further steps the move it from P_i to U_i.
5. For each Q_j, $j = 1,\ldots n$:
   a. create new problem state $p_j = \{A, P_i, T_i, T_{Si}, c, Q_j, U_i\}$
   b. count the lower bound g_j upon the lower bound function:
   c. if $g_j < f^*$, then put p_j into H (**bounding rule**)
6. If $H \neq \emptyset$ then continue at step 2.; otherwise the algorithm is finished and the problem state with the optimal solution is p^*.

During the branch and bound algorithm the problem state with the best solution (called incumbent) is continuously updated and at termination it will hold the global optimum. The algorithm will terminate in finite steps since we repeatedly remove all activities from P as they reach some bounding conditions. In this algorithm the branching and selection rules are the points where the behaviour and efficiency of the algorithm can be altered by selecting different strategies. The branching rule is currently simply to select all the optimizable activities at one and the algorithm uses the best-first selection rule to choose the next problem state to expand, this is one of the most efficient strategies, however, it requires the most memory to store the active (not yet expanded) problem states. It was one of the main reasons that a parallel branch and bound algorithm and a supporting distributed environment were also developed as it is described in the following section. As we are only in the early phase of implementation the measurement of algorithm performance with different strategies is a subject of future work.

VI. SUMMARY

In this paper we showed how can be the optimal resource allocation in a project determined with our RALL-OPT method if a feasible solution is given. On the other hand we described that if the uncertainty of the major variables (duration times, cost and resources) is taken into account when scheduling and allocating a project, then the duration times, costs and resources can be estimated more accurately. The accuracy of the estimation could further be improved if the competence of the human resources is taken into consideration. This way the total project time, total cost and total demands on resources can be determined more accurately, too. Furthermore, as we have more and more information available about the major variables (distribution function of duration time of activities, demands of variable costs and resources or human competences) the more accurate estimation of the resource allocation can be defined.

We also introduced how did we adapted the branch and bound frame method to the optimal solution search within the RALL-OPT method. Supporting this new algorithm we also implemented a distributed environment based on the parallel version of the mentioned B&B algorithm that can utilize spare cycles of networked workstations, used e.g. within a company or project team, to carry out the necessary computations. This system presents a cost effective solution for distributed computing, thus companies or research groups can benefit from parallel computation without investing in expensive multi-processor computers or servers.

VII. APPENDIX

TABLE I

RUN-TIME OF RALL-OPT PROGRAM IN ONE COMPUTER

Number of Activities	30	50	80	100	130	150	180	230	250	310	340	380	390	430	440	480	550	1000	1635	5212
Measure 1 (ms)	15	22	28	32	36	40	42	44	45	50	52	54	58	62	78	120	188	342	812	4524
Measure 2 (ms)	14	21	27	31	35	39	41	43	44	49	51	53	57	61	76	114	157	335	805	4507
Measure 3 (ms)	12	20	27	30	35	39	40	43	43	49	51	52	57	60	76	109	152	335	801	4501
Measure 4 (ms)	12	20	26	30	34	38	40	42	42	48	50	52	56	60	75	108	152	334	785	4421
Measure 5 (ms)	12	19	26	30	34	38	39	42	42	47	50	52	56	58	75	107	151	330	775	4321
Measure 6 (ms)	11	19	26	29	34	37	39	40	41	47	49	51	56	57	74	107	151	329	771	4212
Measure 7 (ms)	10	18	26	29	33	37	39	40	41	46	49	50	55	57	72	106	142	327	770	4201
Measure 8 (ms)	10	18	24	29	32	36	38	39	40	46	48	49	54	55	68	102	140	325	765	4136
Measure 9 (ms)	10	17	24	28	32	36	37	39	39	46	48	48	54	54	64	102	138	320	764	4102
Measure 10 (ms)	10	17	24	27	30	35	37	38	39	45	47	47	53	54	64	101	137	315	760	4024
Mean	**11,60**	**19,10**	**25,80**	**29,50**	**33,50**	**37,50**	**39,20**	**41,00**	**41,60**	**47,30**	**49,50**	**50,80**	**55,60**	**57,80**	**72,20**	**107,60**	**150,80**	**329,20**	**780,80**	**4294,90**
Std. Deviation	1,78	1,66	1,40	1,43	1,78	1,58	1,62	2,05	2,01	1,64	1,58	2,25	1,58	2,90	5,09	5,83	14,81	7,91	18,84	185,01
Rel. Std. Deviation	**0,15**	**0,09**	**0,05**	**0,05**	**0,05**	**0,04**	**0,04**	**0,05**	**0,05**	**0,03**	**0,03**	**0,04**	**0,03**	**0,05**	**0,07**	**0,05**	**0,10**	**0,02**	**0,02**	**0,04**

TABLE II

SPEED-UP EFFECT. (DECREASING THE RUN-TIME OF RALL-OPT PROGRAM WITH DIFFERENT NUMBER OF COMPUTERS IN NETWORK)

Number of activities	Number of computers				
	1	2	5	10	20
30	100%	67%	49%	20%	21%
50	100%	104%	38%	16%	22%
80	100%	96%	21%	16%	16%
100	100%	52%	42%	18%	11%
130	100%	85%	41%	10%	13%
150	100%	108%	29%	17%	6%
180	100%	64%	46%	29%	12%
230	100%	118%	50%	30%	8%
250	100%	69%	41%	20%	14%
310	100%	52%	23%	29%	14%
340	100%	97%	42%	16%	10%
380	100%	101%	49%	10%	6%
390	100%	108%	23%	25%	12%
430	100%	104%	29%	15%	9%
440	100%	77%	25%	15%	13%
480	100%	81%	45%	28%	12%
550	100%	62%	56%	26%	12%
1000	100%	87%	30%	13%	12%
1635	100%	98%	21%	27%	6%
5212	100%	95%	48%	19%	14%
Mean	**1,00**	**0,86**	**0,37**	**0,20**	**0,12**
Std. Deviation	**0,00**	**0,20**	**0,11**	**0,06**	**0,04**

REFERENCES

[1] **Abbasi G. Y., Mukattash A. M.** Crashing PERT networks using mathematical programming. International Journal of Project Management, Vol. 19, Issue 3, pp. 181-188., April 2001

[2] **Beard C. C., Frost V. S.** Prioritized Resource Allocation for Stressed Networks. IEEE/ACM Transactions on Networking, 2001; Vol. 9., No. 5, pp. 618-633., 2001

[3] **Beard C.** Dynamic agent based prioritized resource allocation for stressed networks., Ph.D. dissertation, University of Kansas, Lawrence, 1999

[4] **Bitran G.R., Hax A.C.** Disaggregation and Resource Allocation Using Convex Knapsack Problems with Bounded Variables. Management Science, 1981;Vol. 27, No. 4., pp. 431-441., 1981

[5] **Brucker P., Knust S.** Lower bounds for resource-constrained project scheduling problems. European Journal of Operational Research, Vol. 149, pp. 302-313., 2003

[6] **Chien S. A., Gratch J. M., Burl M. C.** On the Efficient Allocation of Resources for Hypothesis Evaluation: A Statistical Approach. IEEE Trans. Pattern Analysis and Machine Intelligence, 1995; 17(7), pp. 652-665., 1995

[7] **Choueiry B. Y., Faltings B.** A Decomposition Heuristic for Resource Allocation. ECAI 94. 11th European Conference on Artificial Intelligence, Published in 1994 by John Wiley & Sons, Ltd.

[8] **Choueiry B. Y., Faltings B.** Interactive Resource Allocation by Problem Decomposition and Temporal Abstractions., in Current Trends in AI Planning., eds., C. Bäckström and E. Sandewall, 1994;7-104, IOS Press, Amsterdam., pp. 7-104, 1994

[9] **Chrétienne P., Sourd F.** PERT scheduling with convex cost functions. Theoretical Computer Science, Vol. 292, pp. 145–164., 2003

[10] **Cormen T., Leiserson C., Rivest R.** Introduction to Algorithms. The MIT Press, 1990. ISBN 0-262-03141-8 (MIT Press), ISBN 0-07-013143-0 (McGraw-Hill), 1990

[11] **Cung V., Dowaji S., Le Cun B., Roucairol C.** The Outcome of a Know-how: a Branch-and-Bound Library. PRiSM laboratory, University of Versailles, 1994

[12] **Dawson R. J., Dawson C. W.** Practical proposals for managing uncertainty and risk in project planning. International Journal of Project Management, Vol. 16, No. 5, pp. 299-310, 1998

[13] **de Bruin A., Kindervater G.A.P. Kindervater, and Trienekens H.W.J.M. Trienekens.** Parallel Branch and Bound and Anomalies. No 95 in Discussion Paper from Erasmus University Rotterdam, Faculty of Economics, No. 95, 1989

[14] **DeCoster D. T.** PERT/Cost - The challenge. Management Services, pp. 13-18., May-June 1969

[15] **Dodin B. M., Elmaghraby S. E.** Approximating the criticality radices of the activities in the PERT network. Management Science, Vol. 31, pp. 207-223., 1985

[16] **Fatemi Ghomi S. M. T., Rabbani M.** A new structural mechanism for reducibility of stochastic PERT networks. European Journal of Operational Research, Vol. 145, Issue 2, pp. 394-402., 1 March 2003

[17] **Fatemi Ghomi S. M. T., Teimouri E.** Path critical index and activity critical index in PERT networks. European Journal of Operational Research, Vol. 141, Issue 1, pp. 147-152., 16 August 2002

[18] **Frank M., Bugacov A., Chen J., Dakin G., Szekely P., Neches B.** The marbles manifesto: A definition and comparison of cooperative negotiation schemes for distributed resource allocation. In Proceedings of the 2001 AAAI Fall Symposium on Negotiation Methods for Autonomous Cooperative Systems, 36-45. American Association for Artificial Intelligence, pp. 36-45.,November 2001

[19] **Freeman J. B. Jr.** PERT/cost - An informal approach to fundamentals. Management Accounting, pp. 19-26., April 1966

[20] **Garey M. R., Johnson D.S.** Computers and intractability. A guide to the theory of NP-completeness. W.H. Freemann and company, New York, 1979

[21] **Ginis R.** Optimal Distributed Resource Allocation. Thesis. In Partial Fulfillment of the Requirements for the Degree of Master of Science, California Institute of Technology, Pasadena, California, 1999

[22] **Haslum P. and Geffner H.** Heuristic Planning with Time and Resources. European Conference on Planning, 2001

[23] **Heilmann R.** A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags. European Journal of Operational Research, Vol. 144, pp. 348-365., 2003

[24] **Ibaraki T., Katoh N.** Resource Allocation Problems: Algorithmic Approaches. MIT Press, 1988. ISBN 0-262-09027-9., 1988.

[25] **Kamburowski J.** New Validations of PERT Times. International Journal of Management Science, Vol. 25, No. 3, pp. 323-328, 1997

[26] **Kamburowski J.** On the Minimum Cost Project Schedule. International Journal of Management Science, Vol. 23, No. 4, pp. 463-465., 1995

[27] **Kosztyán Zs. T., Bencsik A., Mátrai R.** Handling the Deterministic and stochastic time, resources and costs in project management and logistics. microCAD 2004, Miskolc, Hungary

[28] **Lau H-S, Somarajan C.** A proposal on improved procedures for estimating task-time distributions in PERT. European Journal of Operational Research, Vol. 85, pp. 39-52, 1995

[29] **McKeown G. P., Rayward-Smith V. J. and Rush S. A.** Advances in Parallel Algorithms, Parellel Branch-and-Bound,. IOS Press, Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands, 1994;12-150.

[30] **Modi P. J., Shen W. M., Tambe M., Kulkarni S.** A dynamic distributed constraint satisfaction approach to resource allocation. In Principles and Practica of Constraint programming, 2001

[31] **Mummolo G.** Measuring uncertainty and criticality in network planning by PERT-path technique. Pergamon International Journal of Project Management, Vol. 15, No. 6, pp. 377-387, 1997

[32] **Neumann K., Zhan J.** Heuristics for the minimum project duration problem with minimal and maximal time lags under fixed resource constraints. Journal of Intelligent Manufacturing 6, 1998;145-154.

[33] **Paige H. W.** How PERT/Cost helps the manager. The Harvard Business Review, pp. 87-95., November-December 1963

[34] **Pleguezuelo R. H., Pérez J. G., Rambaud S. C.** A note on the reasonableness of PERT hypotheses. Operations Research Letters, Vol. 31, Issue 1, pp. 60-62., January 2003

[35] **Pontrandolfo P.** Project duration in stochastic networks by the PERT-path technique. International Journal of Project Management, Vol. 18, pp. 215-222., 2000

[36] **Póta Sz.** Design and implementation of a distributed problem solving information system., Master's Degree Thesis, University of Veszprém, Hungary 2002

[37] **Póta Sz., Kosztyán Zs. T.** Implementation of Deterministic and Stochasitc Optimal Resource Allocation in a Distributed System. microCAD 2004, Miskolc, Hungary

[38] **Ravindra K. Ahuja, Thomas L. Magnant, James B. Orlin**, Network Flows., PRENTICE HALL, Upper Saddle River, New Jersey 07458 1998, pp 24-46, 108-116, 135-154, 177-196, 469-173, 520-528

[39] **Ross W. R.** Management acceptance: Potential barrier to PERT/cost implementation. Management Accounting, pp. 58-61., January 1968

[40] **Ross W. R.** PERT/cost. Management Accounting, pp. 47-51., April 1967

[41] **Scholl A., Krispin G., Klein R., Domschke W.** Besser beschränkt, Clever optimieren mit Branch and Bound**.** 1997; c't, Heft 10., 1997

[42] **Valls V., Quintanilla S., Ballestin F.** Resource-constrained project scheduling: A critical activity reordering heuristic. European Journal of Operational Research, Vol. 149, pp. 282–301., 2003

[43] **The ARIS Process Platform home page:** http://www.aris.com

[44] The Jini technology homepage, http://www.jini.org

[45] **Guide to the Expression of Uncertainty in Measurement.** International Organization for Standardization 1993

Using Service-oriented Architectures towards Rights Management interoperability

[1]Carlos Serrão, [1]Miguel Dias and [2]Jaime Delgado

[1]ADETTI/ISCTE, Ed.ISCTE, Av. Das Forças Armadas, 1600-082 Lisboa, Portugal
[2]UPC, Dept. AC, DMAG, Campus Nord Mòdul D6, E-08034 Barcelona, Spain
[1]{carlos.serrao, miguel.dias}@iscte.pt, [2]jaime.delgado@ac.upc.es

***Abstract*— In a World where all forms of digital content are growing at an increasing rate, there are some issues that need to be addressed. Intellectual Property Rights (IPR) is one of the most crucial and important one. If in the analogue World the IPR issues are addressed in a fairly well manner, the same doesn't apply to the digital one. Users are permanently confronted with this important dilemma – to be or not to be a pirate. The choice is not obvious or trivial because it implies choosing between something which freely available on a P2P network or opting by paying an amount of money to get the same content, that is protected and won't work everywhere. In this paper, we will provide some information about the problems arising from the fact that protected content has obtrusive limitations and we will present part of a solution that can be used to address the DRM interoperability problems based on Service-oriented architectures.**

***Index Terms*— Digital Content, interoperability, DRM, SoA, web-services, security**

I. Introduction

Content is part of our daily life. Since we wake up in the morning until we go to bed in the night, in a way or another we are always using content. Content has evolved through time. What we consider old fashioned in our time, such as vinyl records or VHS movies were once state of the art. However, content evolved suffering from some external influences and the main was the Information Technology and Communication one. This digital information revolution has influenced decisively the way content is created, captured, modified and used. From this influence content has become more powerful and appealing.

As someone once said, with great power comes great responsibility, this new digital content types can offer much more to the end users with little effort. However, this significant change has also influenced our lives deeply. In this digital information era, bits are can be moved or copy easily – and they are easier to copy than to move (when we refer to moving we are doing an analogy to the physical world where the same object cannot be twice at the same time). This seems a silly property, but in the digital world it is extremely hard to move bits, and extremely easy to copy bits from side to side. The same property applies to digital content. It is fairly easy to copy from side to side. It is not only easy to copy, but also to modify and distribute. This is where the problem starts.

The problem can be put in a fairly simplistic way: content was authored by someone; this person, the content author, has the copyright over content; the author has the right to receive some payment for its work (because it is fair and because it is lawful); if someone violates this right by using content in a non-authorized way, it also brakes the law. The Intellectual Property Rights (IPR) is a well established principle in our society and is also changing in this digital era.

Once content can be easily copied and distributed through digital means, IPR is most of the times violated in a deliberately or non-deliberately manner. IPR, itself has become digital (e-IPR). Several measures have been created to prevent these e-IPR violations, such as copy-protection (CP) and digital rights management (DRM). These measures are effective up to a certain level; however, they clash against the end-users rights by imposing them a certain level of restrictions that are against the most basic user rights. Users should be able to use their legally acquired digital content anytime, anywhere are anyhow.

Modern DRM and CP mechanisms are too obtrusive for end-users. They impose a vertical DRM model over the content the users acquire, meaning that a specific type of DRM-governed content can only be used on a specific device or on a limited set of devices. As a practical example of this situation, a music file legally acquired at the iTunes online music store, can only be played in the iTunes PC music player or in any of the iPod players – if the user would like to play it on a Creative Zen player, it would be impossible [1][3].

This is a well-known issue of today's DRM systems: interoperability between different DRM systems is virtually inexistent. It is a complex problem to address, and it is not only from a technical point of view, but also from the business perspective point of view. This article is mainly focused on the technical aspects of the problem [2].

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 519–524.

II. DRM GENERIC FRAMEWORK

It is possible to identify the main actors and services of a generic DRM framework (Figure 1).

In general, any DRM framework should address the following actors: the End-User, the Content Owner and the Content Provider. This generic DRM framework, which was originally proposed by [8], identifies seven key DRM services: the Content Service (e.g. search for content), the License Service (e.g. issue licenses), the Access Service (e.g. authenticate consumers), the Tracking Service (e.g. produce usage statistics), the Payment Service (e.g. billing), the Import Service (e.g. submit content to the DRM system), and the Identification Service (e.g. reveal abusers) [8].

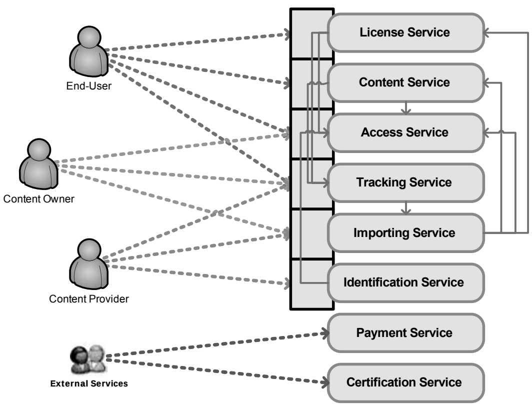


Figure 1 – Generic DRM Framework

While in [8] the authors tend to see DRM in a layered organization to provide interoperability between the different key services of different DRM products, we have a different approach – instead of layers, we tend to see the different DRM key-services as generic distributed services which hide their implementation details from the different DRM actors while exposing publicly their functionalities. This approach could allow the distribution of the different key-DRM services over distributed open-networks, such as the Internet (with proper protection), and DRM key-services to be developed regardless of their own specific and proprietary implementation. This way it can be possible that DRM-services provided by vertical DRM solutions like Windows Media DRM, Helix DNA or even Apple FairPlay to implement a specific interface to expose their basic functionalities to other services. Even new DRM solutions providers can appear in the market and offer their own particular DRM-services without having to implement a complete end-to-end solution. The integration between all the services could provide an interoperable environment.

This approach could also enable the construction of mixed DRM solutions. This idea results from the fact that the some specific and proprietary services from one DRM provider could complement a DRM solution in which such specific service is missing.

III. DRM-SERVICES AS WEB-SERVICES

In computer science, the integration between different software elements provided different vendors, have always represented a huge challenge for system integrators. One set of technologies have emerged to provide a common solution for the problem – middleware.

Middleware is connectivity software that consists of a set of enabling services that allow multiple processes running on one or more machines to interact across a network. Middleware is essential to migrating mainframe applications to client/server applications and to providing for communication across heterogeneous platforms. This technology has evolved during the 1990s to provide for interoperability in support of the move to client/server architectures [18].

These middleware technologies play an important role in the integration of the different applications, including enterprise legacy applications. Technologies such as CORBA, DCOM and RMI have an important role in the integration of applications and services, and recently web-services technology has leveraged these integration technologies to a higher level.

Middleware is computer software that connects software components or applications. It is used most often to support complex, distributed applications. It includes web servers, application servers, content management systems, and similar tools that support application development and delivery. Middleware is especially integral to modern information technology based on XML, SOAP, Web services, and service-oriented architecture.

The same middleware characteristics that can make service oriented architectures helpful for enterprise application integration, dealing with some incompatibility architectures and formats, can also be used to address the DRM interoperability problems.

This paper extends the middleware solution for software and application interoperability problems to a similar scenario in DRM. Each of the DRM services identified previously in the generic DRM framework can be represented as standalone web-services (Figure 2).

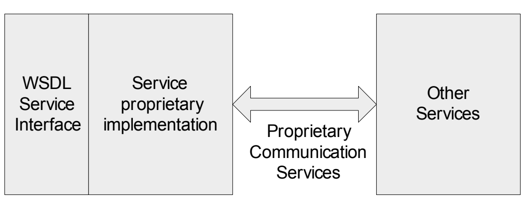


Figure 2 – Implementation of a generic DRM service

Each of these standalone web-services will have its own specific proprietary implementation – that is specific from the supplier of the DRM service – and will have a specific public description of the service available (WSDL). It is envisaged that the middleware interoperability scenario in this case is MOM-based (Message Oriented Middleware). Therefore, the availability of a public web-services interface would provide the facility for services to exchange messages between them in a standardized format (using SOAP).

In more operational scenario it is possible that each of the DRM key-services providers to register themselves on a server repository (UDDI), providing their own description of their services in a normal standard format using WSDL. Each of the registered services is uniquely identified and a set of strong cryptographic credentials (X.509 certificates) are provided to each of these services. This mechanism, already discussed on other papers [7][9][13][14] provides the necessary PKI mechanisms to allow different services to ensure trust each other. This is a critical issue (as we'll describe later) to ensure both authentication and trust between the different DRM services providers.

A scenario like the one which is presented (Figure 3, Figure 4) where there is a multiplicity of Users, Content Providers and License Issuers is not very common in the current days. Nowadays, due to the vertical nature of DRM, the Users all need to share the same device, the Content Providers all need to supply the same type of content format and there can be only one specific License Server. The presented scenario is only possible, because there is a generic WSDL interface provided by each of the License Servers, exposed publicly that allow first the Content Provider to identify which is the User's device to be able to supply the appropriate content type and to be able to create a specific license for that user and content on the appropriate License server, and second it will allow the device to contact the License Server to get the content license using a standardized manner. Although the example provided is very specific for License services it can be extended to all the DRM key-services presented before.

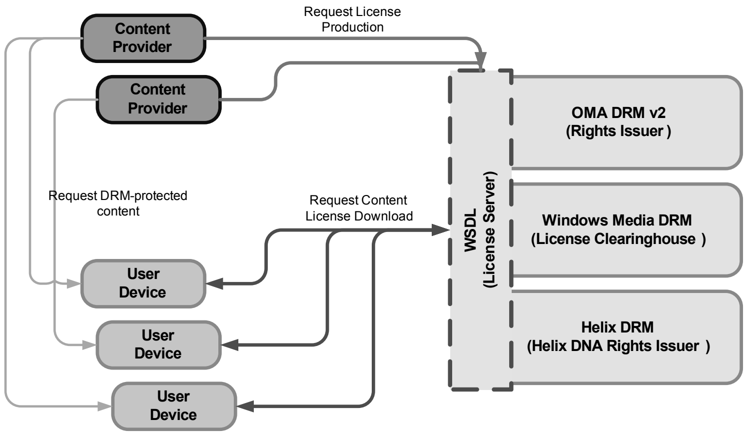


Figure 3 - Defining a generic WSDL interface, common to all proprietary DRM services is important because it allows the coexistence of several non-vertical applications

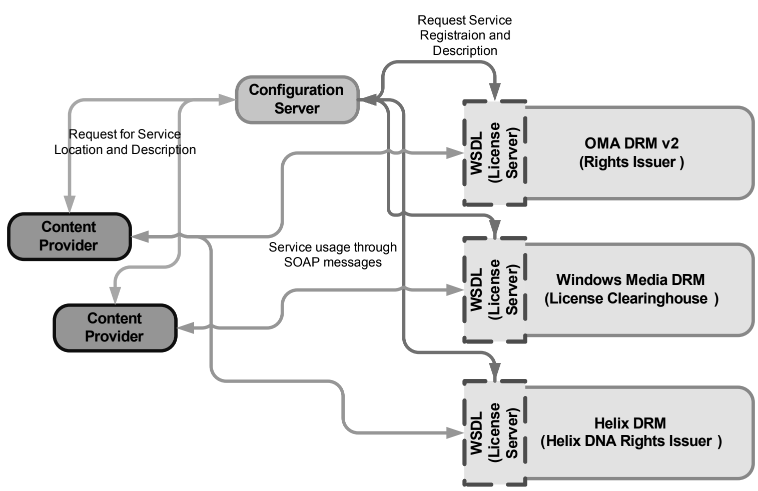


Figure 4 - DRM services registration and description on a registration server. When DRM services are requested, the registration server can provide to the client both the description of the service and the service credentials

This process will require that each of the DRM services suppliers to publish their own web-services description and make them available so that they can be found and used.

IV. DRM INTEROPERABILITY USING A SERVICE ORIENTED ARCHITECTURE

While using the same approach that has been followed by enterprise application integration, it is possible to achieve similar levels of DRM interoperability.

Consider a scenario where a user acquired a DRM-governed piece of content on AStore (AStore is a fictional online content provider store) and wants to be able to play it on its favorite media player. In the current days, this operation is almost impossible to achieve, because they use different, incompatible and non-interoperable DRM systems. However, if we consider a service-oriented approach, like the one that is proposed in this paper, a sequence of operations could be outlined from it:

- The media player, while trying to use the AStore DRM-governed content, would analyze the content and check for the type of DRM-governance that has been applied to the content;
- The player would then contact its local DRM broker (a specific DRM module installed on the end user machine) asking for permissions to render the DRM-governed content;
- The DRM broker would then contact an UDDI server to request information about the DRM services that are requested to render the DRM-governed content;
- UDDI server would then verify the existence of such service and would return the necessary information to the client's DRM broker – in this information pack is the URI of the DRM service (and the WSDL of the DRM service to be used, if available);
- The DRM broker then contacts the DRM service and loads its public available WSDL. The WSDL contains the description of the functions provided by the service as well as the means to use such functionalities;
- The DRM broker can now contact the DRM-service, using the appropriate DRM-service function (or functions) to get the necessary information to render the content (licenses and keys);
- The DRM broker would then allow the media player to render the content.

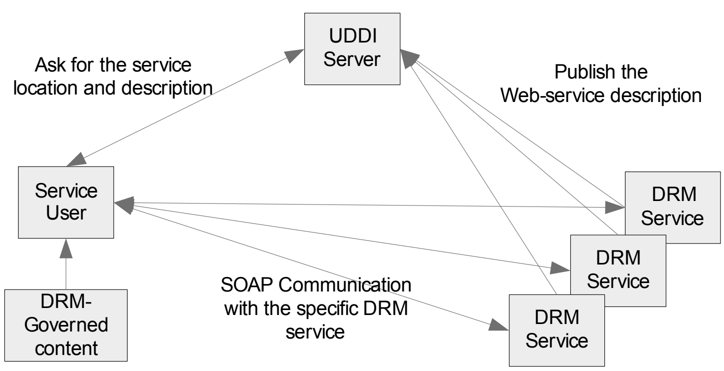


Figure 5 – A scenario with interoperable DRM services using the service-oriented approach (SOA) and Web-services

This is in fact a possible and simplistic scenario, where the presented approach can be used as an interoperability provider.

This SOA scenario has not only advantages for content end-users but also for content providers. Content providers could abstract from the DRM system that will govern their content, and from having to produce a multiplicity of different versions of the same content targeting different platforms and different devices.

This is, however a simplistic high-level conceptual approach to the DRM interoperability problem, since many other issues need to be considered while handling this problem. However, the same SOA approach that we present in the paper could play an important role. Next we will present a list of these issues:

- Content format: content format plays an important role in DRM. The player that is trying to render the content should be able to read it and to understand it in order to take the appropriate actions. In the presented scenario, the media player should be able to read specific tags or markers inside content that signal that content is protected, it's identification and probably the name or identification of the DRM system used;
- Security issues: it is also important to notice that from a security point of view, there are some issues that need to be taken into account. It is imperative that all the communication between the different actors of the system to occur in a secure and authenticated channel. It is also important that the end user media player (user) could be authenticated to the specific DRM service in order

to use the service functions that are available. Moreover, it may need extra authentication credentials to validate a user that is trying to download a license to use some content. Some of the authentication mechanisms used can even be proprietary, or they may be services provided by a different DRM service;

- Rights interpretation: rights can be expressed in many ways, and they may differ from DRM to DRM. It is therefore important that either the media player or the client-side DRM broker to be able to interpret different rights expression languages or capable of translating between those languages. Therefore, there may exist services capable of providing either this interpretation languages or this translation services;
- Content protection: different DRM-governed contents may come with different content protection technologies (using different algorithms, for instance). It is important for the media player to know which specific algorithm has been used to protect the content, so that the inverse operation can be performed and the content used. An additional problem to this is the fact that the media player may not actually have an implementation of such algorithm and therefore may need to obtain it externally using a different DRM service.

It is therefore established that the growing vertical DRM world could benefit greatly from a SOA approach in terms of interoperability.

This SOA approach is being developed on an open-source project called OpenSDRM [7] and has already been tested in terms of adaptation to several heterogeneous scenarios. It was and is currently being used in several projects that handle the most variety of content types, content protection technologies and business models [17]. The platform started being defined and developed for the MOSES [5] project, which handled digital music content in MPEG-4 format and the super-distribution business model. Meanwhile, the OpenSDRM framework was also used in another two projects: WCAM [11] which handles Motion JPEG2000 video-surveillance data and MediaNet [12] which targets the Home Networking environment and the MP3 music files. OpenSDRM was also used in a European Space Agency project [9][10] to control the user rights to access JPEG2000 Earth Observation Products.

The next major challenge for this DRM framework will be the upcoming integration with a peer-to-peer, massive digital content distribution platform [15]. This platform named will be seamlessly integrated and allow for the transparent dissemination of DRM protected content over peer-to-peer networks. This could be regarded as a new direction for research and development, given the sudden apparent recent interest in some of the most powerful players in the content industry business and even from unlikely fields such as space exploration [6]. Indeed the current misconception regarding the utilization of peer-to-peer networks as illicit and condemnable is slowly fading, since they represent an important distribution channel for content. It is however the evolution into the interoperable world, made possible with the developments in DRM approached in this paper, that is making these new paths possible.

V. Conclusions

As it was demonstrated in this paper, the advent of Web Services has had a crucial impact on the Internet, having clearly taking it towards the advent of the Web 2.0 paradigm. In the days of the Web 2.0, technologies such as Web Services and XML allow for an effective separation of content format and content meaning [4]. This created a web scattered with micro-content islands, ready to be accessed by virtually anyone. The appearance of SOA paradigms was only a natural development in this order of events, opening the way to more specialized and independent digital service providers.

In this context, also the distributed programming field incurred in significant changes, namely in the ease of deployment, simplicity of development and lightweight functionalities. To this effect normally cumbersome, proprietary and non-interoperable DRM systems can greatly benefit from these new paradigms in web application development. Not only due to those [later] characteristics per se, but also due to the fact that they make for an overall handicapped user experience, which has been gradually degrading to the point where the Internet content consumer community has been questioning the effectiveness of DRM systems and even their core purpose. As a means of breaking out of this general wave of mistrust and activist like attitude towards DRM as a digital rights management solution, open, interoperable and standards compliant philosophy need to be created.

This new SOA approach effectively contributed as a whole to the great interoperability achieved in this DRM system as well as to the easy and straightforward adaptability to different implementation scenarios, and thus to a possible increase of enjoyable user experience.

References

[1] RUSSINOVICH Mark, "Sony, Rootkits and Digital Rights Management Gone Too Far", Mark's Sysinternals Blog, October 31, 2005, http://www.sysinternals.com/blog/2005/10/sony-rootkits-and-digital-rights.html

[2] MARSON Ingrid, “Sony settles 'rootkit' class action lawsuit”, CNET News.com, December 29, 2005, http://news.com.com/Sony+settles+rootkit+class+action+lawsuit/2100-1002_3-6012173.html

[3] LAMONICA Martin, “New open-source license targets DRM, Hollywood”, CNET News.com, January 18, 2006, http://news.com.com/New+open-source+license+targets+DRM%2C+Hollywood/2100-7344_3-6028284.html?tag=nl

[4] MACMANUS Richard, PORTER Joshua, “Web 2.0 Design: Bootstrapping the Social Web - Web 2.0 for Designers”, Digital

Web Magazine, May 4, 2005, http://www.digital-web.com/articles/web_2_for_designers
[5] MOSES project web-site, http://www.ist-moses.org
[6] NASA Visible Earth, http://visibleearth.nasa.gov/
[7] Serrão C., "Open Secure Infrastructure to control User Access to multimedia content", WEDELMUSIC2004, September 2004, Barcelona
[8] Michiels, S., Verslype, K, et all, "Towards a software architecture for DRM", 5th ACM workshop on Digital rights management, 2005
[9] Serrão C., Serra A., Fonseca P., Dias M., "A Method for Protecting and Controlling Access to JPEG2000 Images", SPIE 2003 Annual Meeting, San Diego, California, United States of America, August 2003
[10] HICOD2000 project web-site, http://www.hicod2000.org
[11] WCAM project web-site, http://www.ist-wcam.org
[12] MediaNet project web-site, http://www.ist-medianet.org
[13] Serrão, C., Dias, M., Delgado, J., "Using Web-Services to Manage and Control Access to Multimedia Content", ISWS05 - The 2005 International Symposium on Web Services and Applications, Las Vegas, USA, 2005
[14] Serrão, C., Dias, M., Delgado, J., "Using ODRL to express rights for different content usage scenarios", ODRL2005 – 2nd International ODRL Workshop 2005, 7-8 July, 2005
[15] Rosenblatt, B., 'Integrating DRM with Peer-to-Peer networks: Enabling the future of online content business models', Whitepaper GiantSteps – Media Technologies Strategies, November 2003
[16] Einhorn, M., Rosenblatt, B., 'Peer-toPeer Networking and Digital Rights Management: How Market Tools can solve Copyright Problems', Policy Analysis No. 534, Cato Institute, February 2005
[17] "Open Digital Rights Language (ODRL) Version 1.1", W3C Note, September 2002, http://www.w3.org/TR/odrl/
[18] Carnegie Mellon Software Engineering Institute, "Definition of Middleware", http://www.sei.cmu.edu/str/descriptions/middleware.html, as visited in 20-10-2006

Corrective Actions at the Application Level for Streaming Video in WiFi Ad Hoc Networks

E. M. Macías, A. Suárez, and J. Martín
Grupo de Arquitectura y Concurrencia, Departamento de Ingeniería Telemática
Universidad de Las Palmas de G. C.
Campus Universitario de Tafira, 35017 Las Palmas de G. C. (Spain)

Abstract—**Efficient video streaming in a mobile ad hoc network (MANET) is a challenging problem due to the dynamic nature of the network that leads to high bit error rates, unpredictable delay, jitter, throughput and packet delivery ratios and frequent short, intermittent and long-term link failures. Despite the MANET research community's efforts, there are still open problems. For example, protocols and mechanisms that hide these issues to the video streaming applications users are still in early stages. However, these applications must tolerate transparently the dynamic behavior of the network and be able to progress in presence of disconnections. In practice, this is the exception rather the rule. In this paper, we present a multilayer cooperative solution to detect disconnections and reconnections between a video streaming server and a client and we propose corrective actions at the application level. With our transparent approach to the user, the video streaming sessions can tolerate frequent long and short disconnections and use more efficiently the shared wireless bandwidth.**

I. Introduction

Achieving multimedia communications over MANET pose many challenges [1] and profitable business [2]. Video streaming is a very useful technique for devices with low storage capacity such as mobile phones and *Personal Digital Assistants* (*PDA*) that use cellular [3], WiFi [4] or WiMax [5] wireless communication technologies. Among others, the following situations degrade the video streaming performance on MANET: i) interruption in packet delivery when a link breaks (e.g. sender, receiver or intermediate node goes out of coverage or their batteries go down), ii) efficient alternative path discovering without degrading jitter [6], iii) high error rates due to the multipath fading, iv) limited bandwidth combined with variable network latency [7]. The efficient election of the transport or application level protocol for video streaming and the efficient control of intermittent wireless channel disruption are also very important issues.

Usually *User Datagram Protocol* (*UDP*) is used to transmit live streaming video [8]. Over UDP, the couple *Real-time Transport Protocol* (*RTP*) and *Real-time Transport Control Protocol* (*RTCP*) is used for real time streaming video [9]. The server continuously sends frames and the client usually does not pause the streaming. A path break implies the server continues with the frame transmission but the client will not receive any frame (data is lost and the bandwidth and battery are not used efficiently).

The persistent version of *HiperText Transfer Protocol* (*HTTP*) can support streaming for *Video on Demand* (*VoD*) so a client sends a request and gets a response, and then sends additional requests and gets additional responses without *Transmission Control Protocol* (*TCP*) connection release. *Real Time Streaming Protocol* (*RTSP*) can use any of the above protocols for transmitting video data and TCP client commands to control the user session on the server.

Although TCP reliability mechanism will retransmit the missing data, the TCP socket will become invalid to the server or the client if an abort occurs and with high probability the user session will end abruptly. Aborts primarily occur when data goes unacknowledged for a period of time that exceeds the limits on retransmission defined by TCP. Other causes for an abort include a request by the application, too many unacknowledged TCP keepalive probes, receipt of a TCP reset packet and some types of network failures reported by the IP layer. We do not consider the improved versions of TCP explained in [10] because of they require: modifications to existing TCP (e.g *TCP-F* and *split-TCP*), more bandwidth and power consumption during a path failure (*TCP-ELFN*), dependency on a particular routing protocol to improve its performance (*TCP-Bus*), addition of layers to the TCP/IP protocol stack (*ATCP*). We do not also consider the protocols overviewed in [11] due to their performance is not well enough [12].

Cross-layer techniques have been applied to solve the above challenges, for example in [13] it is proposed the adaptation of the retry limit parameter at the 802.11 *Medium Access Control* (*MAC*) level to avoid triggering the TCP congestion control mechanism during short-term link failures (it is not appropriated for long-term disruptions). We consider a multilayer cooperative solution: a particular efficient network routing algorithm, any of the above transport protocols, a mechanism to support short and long-term disruptions for TCP based connections, and an application level software that implements the corrective actions when appropriate to robustly tolerate short and long-term disruptions. In order to consider any kind of video streaming client and server we implement a client agent and a proxy server. In this way we

Research partially supported by the Spanish CICYT (MEC) and European Regional Development Fund (FEDER) under Contract TSI2005-07764-C02-01 and The Canaries Regional Education, Cultural and Sports Ministry and FEDER under Contract PI042004/164.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 525–530.

tested the good performance of our multi-protocol and multi-client and server solution for *WiFi* [14] ad hoc networks.

The rest of the paper is organized as follows: section 2 is devoted to discuss the related work. Section 3 reviews the software architecture. Section 4 presents the corrective actions. In section 5 we describe some experimental results. Finally, concluding remarks are summarized in section 6.

II. Related Work

The distributed and self-organizing nature of a MANET stems from having a routing protocol installed in each wireless node. The major routing protocols for MANET are classified into *on-demand* or *reactive* and *proactive* routing algorithms. The former initiate route discovery only after a path breaks incurring a high cost to establish a new route whereas the latter initiate route discovery early and before the path breaks at the cost of higher routing load.

Proactive protocols show more benefits to send video over ad hoc wireless networks than reactive protocols [15]. Due to the reactive behavior, the delay, jitter, throughput and packet delivery ratio for the communication flow may vary a lot in quantity. We use a proactive protocol named *Optimized Link State Routing Protocol* (*OLSR*) [16] that consists of: i) a neighbor sensing mechanism that detects changes in its neighborhood injecting and receiving HELLO messages periodically, ii) an efficient flooding of control traffic, i.e. OLSR packets injected into the network for the quick reconfiguration of path breaks. All nodes receive the messages and there are not duplicated messages thanks to the use of multipoint relays. This is an important property that favors its use in a wireless network which is by nature prone to mobility of nodes and collisions due to the hidden terminal problem, iii) diffusion of topological information necessary to obtain optimal routes in terms of the number of hops. This information is valid for a period of time so expired information is removed. All the traffic in OLSR is UDP and it is transmitted by broadcast or multicast on port 698.

Ref. [17] proposes a hybrid mechanism that consists of an early warning to a reactive protocol in order to initiate route discovery only when a path is likely to break. With this approach, the authors try to reduce the time to detect the disconnection and find a new path, and also reduce the routing load. The signal strength is used as the preemptive trigger. However, as the authors recognize, this physical parameter is not optimal because the value reported differs among 802.11 cards vendors [18]. Therefore, the signal strength values read from a 802.11 card should not be assumed to be particularly accurate [19].

Ref. [20] presents an architecture for detecting and diagnosing faults in IEEE 802.11 infrastructure wireless networks. One of its contributions is enabling bootstrapping and fault diagnosis of disconnected clients to report information to network administrators and support personnel. This work does not provide any support for disconnected clients during on-going video streaming sessions. On the contrary, we provide corrective actions at the application level as well as detection of disconnected clients.

References [15], [17] and [20] are concerned about providing a route between the client and the server in terms of the quick reconfiguration of paths but they are not concerned in providing solutions in the scope of video streaming sessions when a path can not be established.

Multimedia data replication at several servers in a MANET is proposed in [21]. The client establishes a connection to the nearest server once a data block has been received or it is renewed to the same server if it is still the nearest one. To our knowledge, this approach will not perform well with standard streaming protocols because a new connection for VoD implies starting the streaming from its beginning.

The factors causing the low communication quality of current WiFi ad hoc networks and its derived implications for application development is the main concern of [18]. For example, the authors state that applications must tolerate frequent disconnections and the programmers must define when a link is considered to fail but they do not provide a practical solution. We have programmed a proactive mechanism that detects when a link between the client and the server is not available (there is not an alternative route according to the routing protocol) and we do the corrective actions at the application level described in section 4.

Ref. [22] presents a proactive adaptation to the UDP-based streaming video sent by a fixed server to a mobile client connected to one 802.11b infrastructure wireless network. The adaptation consists of increasing the buffer size on the client to store more frames just before entering the low quality area (termed *trouble spot*) in the hope that the mobile client will exit the trouble spot before the buffer runs out. In this paper we consider not only UDP-based streaming video but also TCP-based. On the other hand, we consider both the client and the server mobiles. In this scenario, path breaks take place frequently and quickly so the proactive adaptation proposed in [22] could not be viable.

III. the Software Architecture

Fig.1.a. shows our target MANET consisting of any number of hops. The server node (S) communicates with the client node (C) via zero or more intermediate nodes (I).

The shaded parts in Fig. 1.b to d are the new software elements we introduce (in the protocol stack) to avoid modifying the client application, the server application and the streaming protocols:

- *olsrd* (OLSR daemon) [23] is an implementation of OLSR. OLSR routes efficiently the packets into the network according to the number of hops between the sender and the client. Due to the time-varying characteristics of the wireless links, olsrd can be configured to calculate the optimal routes defined as the number of attempts by a node on average to

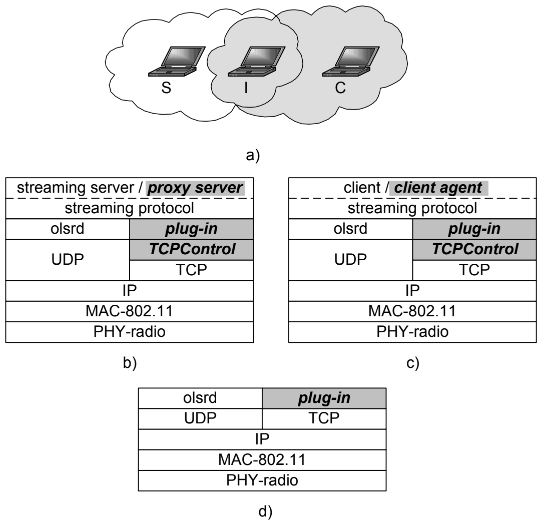


Fig. 1. A two hop MANET. Topology (a), Software architecture: server node (b), client node (c), intermediate node (d).

successfully transmit a packet to a destination, instead of the number of hops. This is a useful behavior since it is important to consider the quality of the link when choosing a path.

- The proxy running on the server and client machines are called *proxy server* and *client agent* respectively. Since both the client and the server are mobile, the proxy server is installed on the wireless node that serves the stream (S), and the client agent on the client node. The proxies are also in charge of detecting if there is, or is not, a path between the server and client to do corrective actions. These corrective actions depend on the type of video streaming being served (VoD or live video) and the type of streaming protocol used to transport the data (RTSP, HTTP or RTP) as we will show in section 4.
- OLSR lets use its optimized flooding mechanism to send information, routing related or not, from the application level using a plug-in. We just use this property to inject user defined packets (OLSR packets type 200) to control the client's and server's availability and to announce the UDP services. The plug-in on the server, client and intermediate nodes conveys to olsrd the information to be sent into OLSR packets type 200 to the MANET. The plug-in on the server and client nodes also communicates to the proxies the OLSR packets type 200 captured by olsrd from the network.

Whenever the communication between the client and the server is possible (via zero or more intermediate nodes), the proxy server receives periodically OLSR packets type 200 from the client agent and viceversa. If the proxy server does not receive at least one of this kind of packet for a while (1 second by default although configurable), this is indicative that the client is disconnected. Similarly, if the client agent does not receive a packet OLSR type 200 from the proxy server (after 1 second by default, also configurable), it becomes aware of the disconnection. The reconnection is detected by the proxy server and the client agent when they receive at least one packet OLSR type 200 from the other one during an interval of 1 second. Appropriate actions are done on both peers when the disconnection or the reconnection are detected. The optimal value for the timeout is difficult to choose: a high value could lead a high delay to detect the disconnection whereas a low value could trigger false alarms, i.e. no packet is received because of network congestion but the proxy server or the client agent wrongly detects a disconnection. The value of 1 second in our experiments gave good results.

- *TCPControl* [24] is the mechanism for transparently detecting TCP connection failures and to create a new TCP connection that avoids the streaming session release.

IV. CORRECTIVE ACTIONS

Table 1 summarizes the actions that the proxies do when they detect disconnections and reconnections (in brackets it is shown the process that does the action), and the benefits of these actions. Irrespective of the streaming protocol, the client agent starts a warning message box on the user's screen when a disconnection or a reconnection is detected and the proxy server ends the session when the disconnection exceeds a period of time. For the streaming protocols built on top of TCP (RTSP and HTTP), the TCP connection between the proxy server and the client agent is closed when a disconnection happens and a new one is created after the reconnection using our TCPControl mechanism. Using RTSP compliant commands such as pause and play, the proxy server pauses or resumes the server. For HTTP or RTP, the server is not paused during the disconnection period but the frames are not forwarded from the proxy server to the client agent to save bandwidth.

V. EXPERIMENTAL RESULTS

We tested the behavior and performance of our software architecture using the topology showed in Fig. 1.a. We are concerned in presenting results that show the benefits of using our corrective actions and the TCP connections management between the proxy server and the client agent. For doing that, we did several experiments that consisted of forcing client's disconnections and reconnections, and evaluating the behavior for RTSP, HTTP and RTP/UDP based streams using or not our proxies based solution. We measured the data volume and the TCP disconnections during a disconnection and show how this is solved with our approach.

TABLE I
CORRECTIVE ACTIONS DURING DISCONNECTIONS AND RECONNECTIONS

Action vs. Protocol	**RTSP**	**HTTP**	**RTP**
Disconnection	Pause the server (PS), warning the	Freeze frames forwarding from	Freeze frames forwarding from

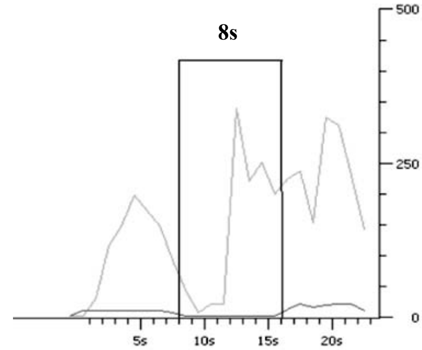
	user (CA), close TCP connection (PS,CA)	PS to CA, close TCP connection (PS,CA), warning the user (CA)	PS to CA, warning the user (CA)
Reconnection	Create TCP connection (PS,CA), resume the server (PS), warning the user (CA)	Create TCP connection (PS,CA), resume frames forwarding (PS), warning the user (CA)	Resume frames forwarding (PS), warning the user (CA)
Total disconnection	End session (PS)	End session (PS)	End session (PS)
Lost frames	Yes (live video), No (VoD)	Yes	Yes
Abrupt ending	No	No	--
Batt. saving	Yes	No*	No*
BW saving	Yes	Yes	Yes

PS: Proxy Server **CA**: Client Agent **Batt.**: Battery **BW**: Bandwidth
* The server is still sending frames but PS does not inject them into the MANET

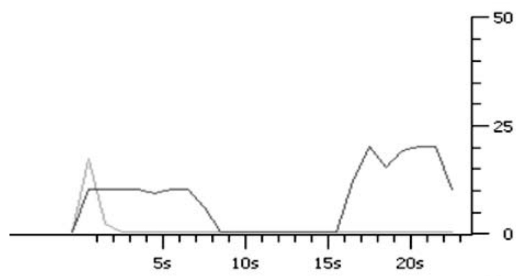
The plug-ins and the proxies were programmed using C and C++ languages respectively for Windows operating system. The server was installed on a Pentium IV at 2.8 GHz with 512 MB and 802.11b compliant. The intermediate node was a Centrino at 1.6 GHz, 512 MB and 802.11b/g. The client node was a Celeron 1.4 GHz, 1024 MB and with a 802.11b/g wireless interface. All the nodes were located in the same room and we added mobility to the network by allowing the client node to be within radio range of the server node via the intermediate node and we also moved the client and the server to make each other out of coverage to test the corrective actions made by proxies and the TCP connections management. We used VLC media player [25], a free cross-platform media player that supports a large number of multimedia formats and it is available for several operating systems, it needs little CPU power and it can be used as a streaming server to stream unicast or multicast in IPv4 or IPv6. We used VLC for serving the video in unicast in IPv4.

A. *RTSP*

Fig. 2.a presents the number of packets per second injected by the server during a VoD RTSP streaming session at a rate of 2 Mbps (green curve) and the OLSR traffic transmitted by the client node including our packets type 200 (red curve). The green curve only shows the multimedia traffic using RTP protocol, i.e. RTSP commands using the TCP connection are not shown in this curve but in Fig. 2.b. We forced a disconnection period of 8 s (about the 8^{th} second until the 16^{th} second). During this time interval, Ethereal tool did not capture OLSR traffic (the red curve falls to 0) since the client is out of coverage. However, Ethereal captured RTP traffic transmitted by the server since the server is not aware of the disconnection period (no proxies were used for this test). As a



a)



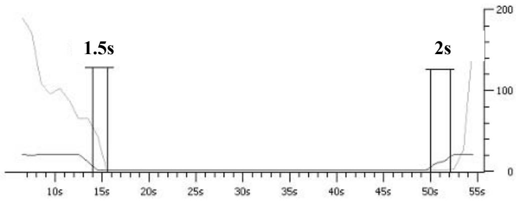
b)

Fig. 2. Behavior during a disconnection and after the reconnection for a RTSP session without corrective actions: RTP traffic from the server (green curve) and OLSR traffic (red curve) from the client (a). TCP traffic between the client and the server (green curve) and OLSR traffic (red curve) from the client (b).

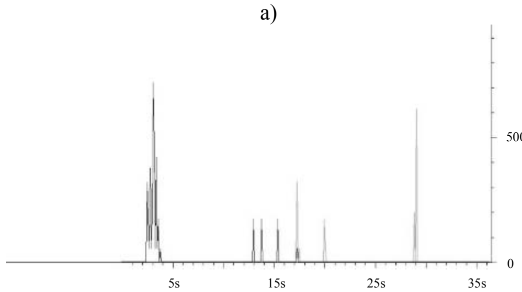
result, about 2 MB are transmitted and lost using RTP protocol since we do not use a proxy server to pause the server.

Fig. 2.b presents the number of packets per second injected by the server and the client using the TCP connection (green curve). As you can see, the TCP connection is lost during the disconnection period (again red curve shows the OLSR traffic injected by the client that falls to 0 during the disconnection period) and it is not recovered after the client's reconnection. As a result, any attempt of the client to use this TCP connection to control the streaming will fail or even the streaming session will end abruptly.

To correct this inefficient usage of the server and the available wireless bandwidth, and to avoid the lost of the TCP connection, we repeated the experiment using the proxies and we forced a higher disconnection period of 35 s (Fig. 3.a). During this period, the server is paused by the proxy server and no frames are transmitted avoiding that the server injects a total of 8.75 MB. As it is shown if Fig. 3.a, the proxy server lasts about 1.5 s to detect and react properly to the disconnection and about 2 s to detect and react to the reconnection. Both values are a bit higher to the theoretical value of 1 second we fix to warn the proxy about a



a)



b)

Fig. 3. Improved behavior using proxies during a disconnection and after the reconnection for a RTSP session: RTP traffic from the server (green curve) and OLSR traffic (red curve) from the client (a). TCP traffic between the client and the server (b).

disconnection or reconnection because it is included the time the proxy server needs to do a corrective action, e.g. sending a RTSP compliant pause or play command to the server. Since we use a proactive protocol to detect them, the detection time is even lower that the one we would obtain using reactive protocols such as *Ad hoc On-demand Distance Vector Routing* (*AODV*) or *Dynamic Source Routing* (*DSR*) [10].

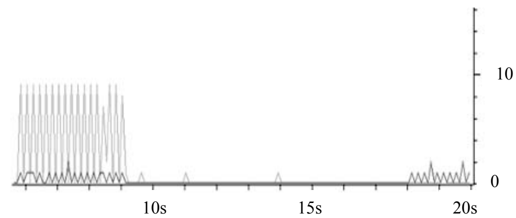
Fig. 3.b shows the behavior of the TCPControl mechanism for the RTSP session. Initially, the port used for the TCP connection between the client agent and the proxy server is 1273 (blue curve). About the second 17, the TCP connection is lost but using TCPControl a new TCP connection over port 1280 is created (violet curve) and the streaming session is resumed and not abruptly ended after the reconnection.

B. HTTP

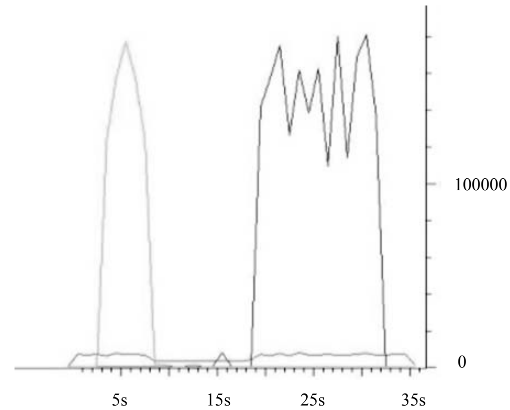
Fig. 4.a presents the number of packets per second injected by the server during a streaming session over HTTP (green curve) and the OLSR traffic transmitted by the client (red curve). During the disconnection period, i.e. the red curve falls to 0, the HTTP based stream is not captured since the TCP connection is lost. After the reconnection, the TCP connection is not recovered so the streaming is stopped and any attempt of the reconnected client to receive the stream will fail. However, using proxies (Fig. 4.b), the TCP connection is reestablished once the client is reconnected (blue curve in Fig. 4.b) using our *TCPControl* software.

C. RTP

Fig. 5.a presents the number of packets per second injected



a)



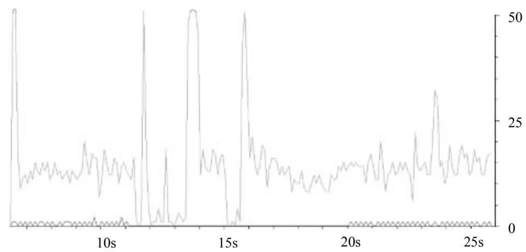
b)

Fig. 4. HTTP based streaming: Behavior during a disconnection and after a reconnection without corrective actions (a). Improved behavior with proxies (b).

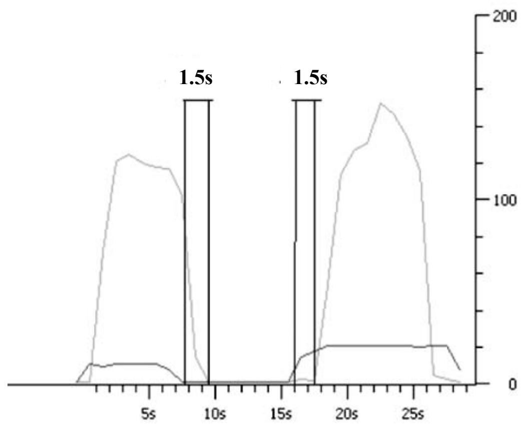
by the server during a streaming session over RTP/UDP (green curve) at a rate of 2 Mbps and the OLSR traffic transmitted by the client (red curve). We forced a disconnection period of 9s (about the 11th second to the 20th second). During this time interval, Ethereal did capture RTP/UDP traffic transmitted by the server since the server is not aware of the disconnection period. As a result, a data volume of 2.25 MB is transmitted using inefficiently the wireless bandwidth. We repeated the experiment using proxies. Fig. 5.b shows the RTP/UDP stream captured by Ethereal (green curve) and transmitted by the server, and the OLSR traffic sent by the client (red curve) both in terms of number of packets injected per second. This time, the proxy server lasts about 1.5s to detect and react properly to the disconnection and the reconnection. During the disconnection period, no RTP/UDP based stream is forwarded by the proxy server to the proxy client and as a result, bandwidth is saved.

D. Percentage of recovered TCP connections

For video streaming based on RTSP and HTTP we forced 25 disconnections between the client and the server and we studied the percentage of TCP connections recovered. This value was 92% (23 successful reconnections). For the two TCP connections lost and not recovered by our TCPControl mechanism, the server's resources allocated for these



a)



b)

Fig. 5. RTP based streaming: Behavior during a disconnection and after a reconnection without corrective actions (a). Improved behavior with proxies (b).

streaming sessions were silently released.

VI. CONCLUSION

This paper discussed about the challenges that a video streaming session faced in a MANET. We proposed some corrective actions at the application layer for different streaming media protocols. This support is done by proxies that detect path breaks and reconnections thanks to the feedback provided by the proactive OLSR protocol. Experimental results showed the convenience of using our software architecture for a better use of the bandwidth and to avoid losing of frames under certain conditions. We are thinking to improve the corrective actions, e.g. store frames during disconnections for HTTP and RTP based streams and give early directions to the user to a better coverage area.

REFERENCES

[1] M.D. Kwong, S. Cherkaoui, and R. Lefebvre, "Multiple description and multi-path routing for robust voice transmission over ad-hoc networks", in *2006 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 262-267.

[2] Y. Tashiro, Y. Yashima, and H. Fuju, "NTT's technologies for next-generation video services", *ACM Computers in Entertainment*, vol. 4 (1), January 2006.

[3] P. Fröjdh, U. Horn, M. Kampmann, A. Nohlgren, and M. Westerlund, "Adaptive streaming within the 3GPPP packet-switched streaming service", *IEEE Network*, vol. 20 (2), pp. 34–40, March-April 2006.

[4] N. Cranley, and M. Davis, "Study of behaviour of video streaming over IEEE 802.11b WLAN networks", in *2006 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications,* pp. 349–355.

[5] O.I. Hillestad, A. Perkins, V. Genc, S. Murphy, and J. Murphy, "Delivery of on-demand video services in rural areas via IEEE 802.16 broadband wireless access networks", in *2006 Proceedings of the Second ACM International Workshop on Wireless Multimedia Networking and Performance Modeling*, pp. 43-51.

[6] H. Fujisawa, H. Minami, M. Yamamoto, Y. Izumi, and Y. Fujita, "Route selection using retransmission packets for video streaming on ad hoc networks", in *2006 IEEE Radio and Wireless Symposium*, pp. 607–610.

[7] J. Ding, C. Lin, and K. Huang, "ARS: an adaptive reception scheme for handheld devices supporting mobile video streaming services", in *2006 IEEE International Conference on Consumer Electronics, Digest of Technical Papers*, pp. 141–142.

[8] G. Cunningham, S. Murphy, L. Murphy, and P. Perry, "Seamless handover of streamed video over UDP between wireless LANs", in *2005 Consumer Communications and Networking Conference,* pp. 284–289.

[9] J. Li, and L. Li, "Research of transmission and control of real-time MPEG-4 video streaming for multi-channel over wireless QoS mechanism", in *2006 1st International Multi-Symposiums on Computer and Computational Sciences*, vol. 2, pp. 257-261.

[10] C. Siva Ram Murthy, and B. S. Manoj, *Ad Hoc Wireless Networks. Architectures and Protocols*. Prentice Hall, PTR, 2004.

[11] G. Yang, L. Chen, T. Sun, M. Gerla, and M.Y. Sanadidi, "Real-time streaming over wireless links: a comparative study", in *2005 Proceedings of 10th IEEE Symposium on Computers and Communications*, pp. 249–254.

[12] M. Li, C. Lee, E. Agu, M. Claypool, and R. Kinicki, "Performance enhancement of TFRC in wireless ad hoc networks", in *2004 Proceedings of the 10th International Conference on Distributed Multimedia Systems*.

[13] S. Lohier, Y. Ghamri-Doudane, and G. Pujolle, "MAC-layer adaptation to improve TCP flow performance in 802.11 wireless networks", in *2006 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 427–433.

[14] IEEE 802.11, The Working Group Setting the Standards for Wireless LANs [Online]. Available: http://grouper.ieee.org/groups/802/11/

[15] J. Karlsson, H. Li, and J. Eriksson, "Real-time video over wireless ad-hoc networks", in *2005 14th International Conference on Computer Communications and Networks*, San Diego, California.

[16] T.H. Clausen, and P. Jacquet. (2003, October). Optimized Link State Routing Protocol, RFC3626, Internet Engineering Task Force (IETF) [Online]. Available: http://ietf.org/rfc/rfc3626.txt

[17] T. Goff, N.B. Abu-Ghazaleh, D.S. Phatak, and R. Kahvecioglu, "Preemptive routing in ad hoc networks", in *2001 ACM SIGMOBILE*, pp. 43-52.

[18] G. Gaertner, and V. Cahill, "Understanding link quality in 802.11 mobile ad hoc networks", *IEEE Internet Computing*, pp. 55-60, January-February 2004.

[19] J. Bardwell. (2004). You Believe You Understand What You Think I Said. The Truth About 802.11 Signal and Noise Metrics [Online]. Available: http://www.connect802.com/download/techpubs/2004/you_believe_D100201.pdf

[20] A. Adya, P. Bahl, R. Chandra, and L. Qiu, "Architecture and techniques for diagnosing faults in IEEE 802.11 infrastructure networks", in *2004 10th Annual International Conference on Mobile Computing and Networking*, pp. 30-44.

[21] S. Ghandeharizadeh, S. Kapadia, and B. Krishnamachari, "Server replication techniques for display of continuous media in mobile ad hoc networks", Computer Science Department, University of Southern California, Los Angeles, CA, Tech. Rep. 2005.

[22] V. Sunderam, J. Pascoe, and G. Tonev, "Reconciling the characteristics of wired and wireless networks: the Janus approach", in *2002 4th Annual International Workshop on Active Middleware Services*.

[23] O L S R . O R G [Online]. Available: http://www.olsr.org/

[24] E. Macías, A. Suárez, J. Martín, and V. Sunderam, "Using OLSR for streaming video in 802.11 ad hoc networks to save bandwidth", *Special Issue of IAENG*, in press.

[25] VideoLAN - Free Software and Open Source Video Streaming Solution for Every OS! [Online]. Available: http://www.videolan.org/

Visual Data Mining of Log Files

Guillermo Francia, III Monica Trifas Dorothy Brown Rahjima Francia Chrissy Scott
Department of Mathematical, Computing and Information Sciences
Jacksonville State University
700 Pelham Road North
Jacksonville, AL 36265

Abstract Data mining is based on a simple analogy. The growth of data warehousing has created mountains of data. The mountains represent a valuable resource to the enterprise. But to extract value from these mountains, we must "mine" for the gold in data warehouses and data marts. Everywhere that there are data warehouses, data mines are also constructed.

Data visualization has the ability to present a great deal of information in a user friendly format. It is well known that humans comprehend visual information much quicker and more efficiently than verbal information. "A picture is worth a thousand words." Successful visualizations can reduce the time it takes to get the information, make sense out of it, and enhance creative thinking. Great strides have been made in the area of computer generated data visualizations in recent years.

This paper discusses visual data mining techniques for analyzing real forensic data.

INTRODUCTION

A. Data Mining

Data is the basic form of information that needs to be collected, managed, mined and interpreted to create knowledge. Discovering the patterns, trends, and anomalies in massive data represents one of the grand challenges of the information age. The more data someone has to handle, the more difficult it is to effectively analyze and draw meaningful conclusions from it. Data mining uses analytic technologies to quickly explore mountains of data and to provide the usable information the user needs. Data mining is a multidisciplinary field drawing upon works from statistics, database technology, artificial intelligence, pattern recognition, machine learning, information theory, control theory, information retrieval, high-performance computing, and data visualization. The aim of data mining is to extract implicit, previously unknown and potentially useful patterns and models from data [8].

Data mining derives its name from the similarities between searching for valuable business information in a large database and mining a mountain for a vein of valuable ore. Both processes require either sifting through an immense amount of material, or intelligently probing it to find exactly where the value resides.

Data mining applications have been shown to be highly effective in addressing important business problems, research activities, and engineering solutions. We expect to see a continuing trend in the building and deployment of data mining and knowledge discovery applications for crucial business and scientific decision support systems.

B. Data Visualization

Data visualization is a powerful tool in the field of Information Science. Information Science, along with the field of Computer Science, saw phenomenal growth after World War II. At this time there was an "information explosion" - an exponential growth of scientific publications and literature due to the many advances in science and technology since the beginning of the twentieth century [12]. Discovering information became the new "Gold Rush". Researchers, professionals, and businesses raced each other to find this precious commodity [12]. Dictionaries state that information science deals with the collection, storage and retrieval of information. Those in the field of Computer Science, however, claim it as "a field of professional practice and scientific inquiry addressing the problem of effective communication of knowledge records [12]".

Visual representation and interaction technologies provide a mechanism allowing the user to see and understand a large amount of information at once. The human mind can indisputably understand complex information received through visual channels. Based on this ability, visual analytics facilitates the methodical reasoning process.

Creating effective visual representations is a labor-intensive process that requires a solid understanding of the visualization pipeline, characteristics of the data to be displayed, and the tasks to be performed. An efficient technique for visual representations must employ cognitive and perceptual principles that can be deployed through engineered, reusable components. Visual representation principles must address all types of data, scale and information complexity, enable knowledge discovery, and facilitate analytical reasoning.

Visual analytics software uses visual representations and interactions to accelerate rapid insight into complex data. Visual representations translate data into a visible form that highlights important features including commonalities and anomalies. These visual representations allow the users to perceive salient aspects of their data quickly. The cognitive reasoning process can be augmented through perceptual reasoning, by visual representations. Thus, the analytical reasoning process becomes faster and more focused.

It is a challenge to create well-constructed visual representations. In the field of scientific visualization, data often corresponds to real-world objects and phenomena. In scientific visualization, the goal is to reproduce these real-world representations as faithfully as computationally feasible. However, most visual analytic problems manipulate

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 531–535.

abstract information; therefore the researcher has to select the best representation for the information.

Visual representations invite the user to explore the data. This exploration requires that the user be able to interact with the data to understand trends and anomalies, isolate and reorganize information as appropriate, and engage in the analytical reasoning process. The analyst gains insight through these interactions.

The design of visual representations of information has been ongoing for centuries. Over the past 20 years, the increasing speed and availability of computers allowed information visualization researchers to create dynamic and interactive computer-mediated visual metaphors for depicting abstract information.

This paper demonstrates the viability of using Log Parser and Mineset in visual data mining. In our endeavor, we have taken real forensic data from two web servers and a desktop computer to generate visual images for data mining and interpretation. This work is part of an on-going project in the area of visualization and management of digital forensic data.

MS LOG PARSER TOOLKIT 2.2

A log is a record composed of log entries containing information about the events occurring within an organization's systems and networks. Previously, logs were used primarily for troubleshooting problems, but logs now serve many functions within most organizations, such as optimizing system and network performance, recording the actions of users, and providing data useful for investigating malicious activity [16].

Log management is vital for organizations. The information contained within these log entries are useful for performing auditing and forensic analysis, supporting an organization's internal investigations, establishing baselines, and identifying operational trends and long-term problems [16].

Filtering through a large number of log entries can be almost impossible. Once log entries are gathered, one needs to sort, aggregate, normalize, and correlate them in order to produce functional data with significant patterns. One such tool that is capable of doing the aforementioned is the MS Log Parser Toolkit 2.2.

The Log Parser tool first appeared in 2000 as a utility to test the logging mechanism of Microsoft's Internet Information Services (IIS). This allowed users to retrieve and display all the fields from a single log file in any of the three text-logging formulas supported by IIS. Since then, as tests became more complex, specifically the filtering through log entries, Microsoft saw an immediate need for a log management tool. Version 2.0 was the first available version outside Microsoft. MS Log Parser Version 2.2, which shipped in January 2005, was designed and engineered with the vision of helping users achieve their data-processing goals in a simple, fast, and powerful way [4].

Log Parser gives you a way to create a data processing channel by mixing and matching input formats and output formats as needed, using a query written in Structured Query Language (SQL). Input formats can be thought of as SQL tables containing data to be processed and output formats as SQL tables that receive the results of the data processing. Thus, the Log Parser contains an SQL-like engine core that is proficient enough in performing input and output processing of web log files [4].

Once logs are gathered, there is a need for further processing in order to make it more perceptible to users. Log data are displayed in a human-readable format for functional reporting or monitoring for anomalies. One of the most exciting features of Log Parser version 2.2 is its ability to automatically generate graphical charts based on the queried information. One can generate dozens of chart types, including bar charts, pie charts, line charts, and more. An example of a 3D bar chart that was generated by Log Parser is shown in Fig.1 [4].

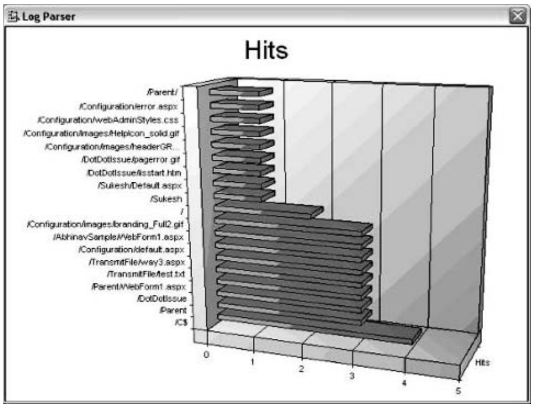


Fig. 1. Using the Log Parser

MINESET

Mineset is a commercial visual data mining product from Silicon Graphics, which was founded in 1982. Their initial focus was to introduce the market to new technologies that would allow users to interact with their data in 3D [13]. Mineset was first released by Silicon Graphics in 1996 primarily as a data mining and visualization product [1]. On October 23, 2003, Silicon Graphics announced an agreement with Purple Insight for the distribution of Mineset Data Mining and Real-Time 3D Visualization Software [15]. Purple Insight is a software and services company. It is the world's premier provider of Visual Data Mining solutions [11]. Silicon Graphics and Purple insight have the same vision in regards to the power of visualization [15].

Mineset makes use of a three tier architecture and is depicted in Fig.2. The first tier includes the Tool Manager and visualization tools. It is called the client tier. The visualization tools use mining algorithms to generate and display data and visual models. The Tool Manager is a graphical user interface that allows users to interface with Mineset. The second tier is called the server tier. This tier includes the Data Mover and the analytical mining engine. The Data Mover extracts data from the third tier, transforms it, and coordinates moving it from one component to the other. The mining tools are used to generate models of the transformed data. These models can be applied to new or previously visualized data. The third tier stores and maintains the user's data. This is called the Data Source tier. The actual data source can be a data file, database, or data warehouse [1].

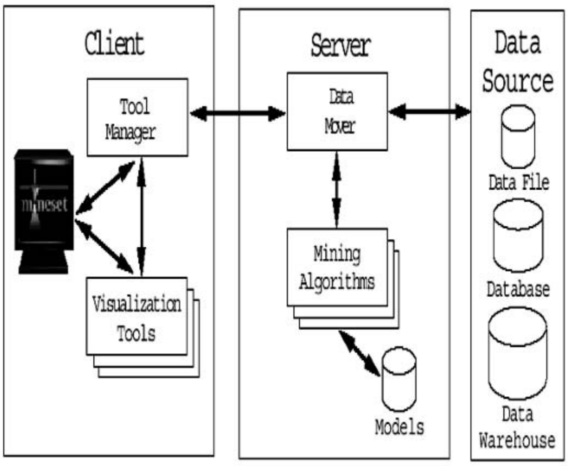


Fig 2 Mineset's 3-tier Architecture [11]

Mineset enables the interactive exploration of data by providing a rich set of visualization tools. The visual tools utilize 2D and 3D visualization capabilities. Mineset includes eight tools, which are called visualizers. These tools are described in the following section.

The *Statistics visualizer* displays statistics in histograms and box plots. It computes and displays summary information for the current dataset (maximum, minimum, median, standard deviation, distinct values, and quartiles). The *Cluster Visualizer* extends the Statistics Visualizer. It shows the differences, for each attribute, between clusters. It places these statistics side-by-side with those for the entire data set, so that the unique features of each cluster can be seen. The *Tree Visualizer* helps analyze data that has hierarchical relationships. It provides an interactive "fly through" capability for examining relationships among data at different levels. The *Map Visualizer* displays data with a spatial component. It allows for visualization of data relationships that exist across geographically meaningful areas. This visualizer lets you visually examine patterns in data that are difficult to detect when the data is shown in 2-D form. The *Scatter Visualizer* displays scatter plots. A scatter plot can have up to eight dimensions (three axe, entity color, entity size, entity rotation, and two independent attributes shown through animation). The *Splat Visualizer* extends the Scatter Visualizer to produce 3-D plots of very large data sets. The *Decision Table Visualizer* allows for distribution of data from a discrete column at multiple levels of a hierarchy. It breaks down the class label according to attribute value. Finally, the *Evidence Visualizer* shows a graphical representation and allows the user to interact with the model by providing what-if-analysis.

Mineset approaches data-mining in many ways, supports a variety of techniques and uniquely combines these with dynamic interactive visualizations, putting more power into users' hands. Mineset provides other unique benefits including: scalability to handle mass amounts of data, Multi-platform - support of wide range of databases and operating systems, an API that makes it extendable and is web deliverable, and visualization tools unique in the industry.

VISUAL DATA MINING RESULTS

This paper presents some of the results of an on-going project in the area of visualization and management of digital forensic data. The following figures were generated as a result of the utilization of Log Parser and Mineset on real forensic log files. Each figure indicates one or more pieces of information that can not be gleaned by merely observing the data in text or tabular format.

Fig 3 depicts the daily file system activity and, in particular, the frequency of file modifications. In forensic analysis, this is a very important tool to elicit information about the days in which the user is mostly busy manipulating files in a computer system. Here is the Log Parser query that generated this chart:

```
SELECT TO_DATE (LastWriteTime) as
LastWriteDate,   COUNT(*) as
WriteFrequency  INTO     C:\\TempChart.gif
FROM    C:/Temp/*.*     GROUP BY LastWriteDate
```

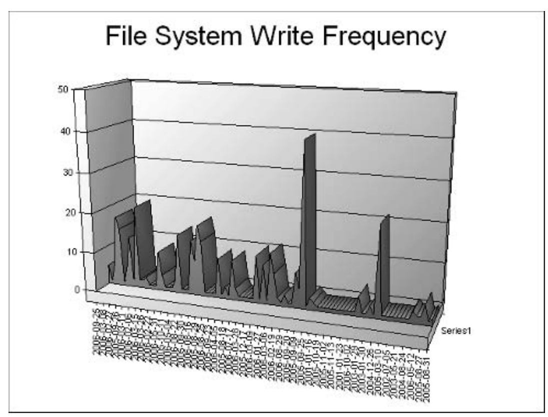


Fig 3. Daily Frequency of File System Modifications

Fig 4 depicts the activity of the registry files during the last 24 hours. This forensic visual aid is very important in identifying the exact location of registry entries that were used during the installation or deletion of system files and application programs. Here is the Log Parser query that generated this chart:

```
SELECT Path, count(*) as [Keys Modified]
    INTO C:\\TempChart.gif FROM HKLM\\SOFTWARE
    WHERE LastWriteTime >=
          SUB(SYSTEM_TIMESTAMP( ),
         TIMESTAMP('0000-01-02','yyyy-MM-dd'))
    GROUP BY Path
```

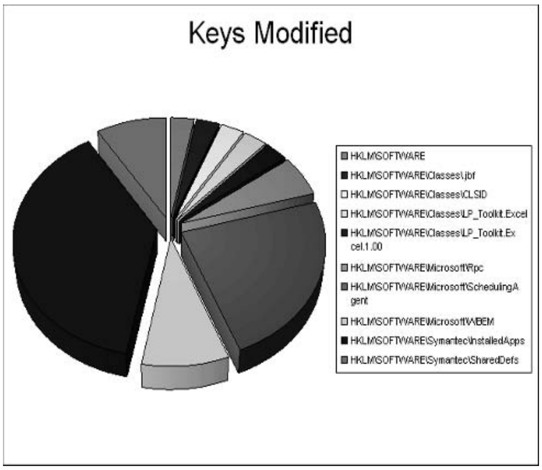


Fig 4. The Frequency of Activity by Registry Path

In Fig 5, the access log file of our main server was used for analysis. The log file contains 18 months of historical data with approximately 2.8 million records. A cursory examination of the figure reveals a significant amount of illegal and dangerous client requests as shown by the status codes 404 (page not found) and 405 (method not allowed). Here is the Log Parser query that generated this chart:

```
SELECT StatusCode, Count(*) as [Status Code Count]
     INTO C:\\TempChart.gif FROM C:\\temp\\access2.log
    GROUP BY StatusCode ORDER BY StatusCode
```

Fig 6 displays the total bad status code counts for each client IP taken from an Apache web server's access log file. The log file, which covers 22 days of continuous operation, contains 2548 records. The visual representation embodies a large amount of forensic information. These non-apparent clues can be garnered through careful inspection. For example, the orange bar represents a total of 101 status codes for a client whose IP number is [redacted].231.117 and the likewise, a total of 50 status codes for the blue bar representing a client having IP number [redacted].24.19. This can be very useful for an astute system administrator in blocking the requests coming from clients which are suspected to have some malicious intentions.

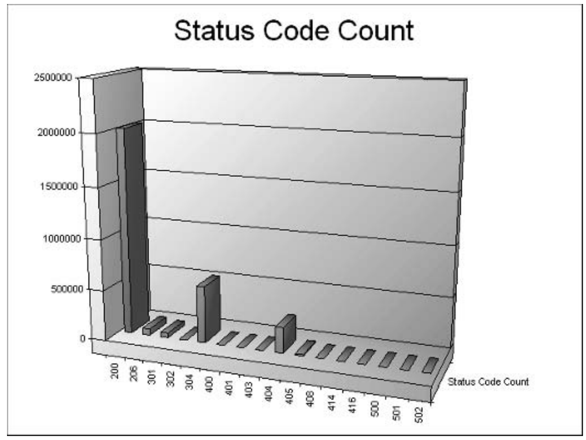


Fig 5. The Status Code Frequency of Server 1

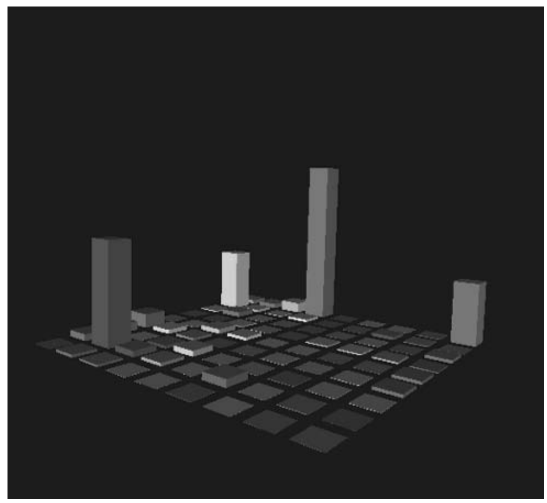


Fig 6. Total Count of Bad Status Codes for each IP Client

In Fig 7, a scatter plot displays the data which is extracted from a security log file in our secondary web server. The log file, which spans 22 days of continuous activity, contains a total of 137,681 records. The figure is a snapshot of an animation and fly-through that we used as an instrument in visual data mining. The most significant result that an analyst can gather from this figure is the strong correlation of the installation of vulnerable application programs with the frequency of attempted break-ins.

CONCLUSION AND FUTURE WORK

We have presented some of the results of an on-going research project in the area of visualization and management of digital forensic data. Our preliminary results clearly indicate the

This page intentionally blank

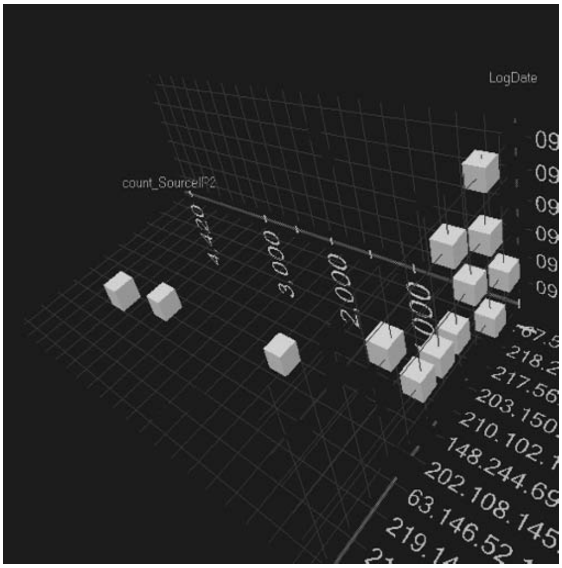


Figure 7. Scatter Plot of a Security Log

viability of using visualization tools in extracting new information from vast amount of data. Such new information may not be perceptible even with very close examination of the data in textual form. Furthermore, it became evident during the course of our research adventure that a proper correlation of multiple log files produces unanticipated results that proved to be very beneficial.

The direction of our future work has been determined and it includes the following:

1) Automate the process of data cleansing of log files.
2) Extend the visual data mining process to include the more advanced data mining algorithms in Mineset.
3) Create a graphical user interface for Log Parser.
4) Extend the data correlation process to include more log files.
5) Provide a system for invoking the Mineset APIs from an application program.
6) Facilitate the web-enablement of the generated charts.

ACKNOWLEDGMENTS

This project is partially funded by the Computing Research Association Committee on the Status of Women in Computing Research (CRA-W) and the National Science Foundation (NSF).

Opinions expressed are those of the authors and are not necessarily of the Foundation.

REFERENCES

[1] C.Brunk, R. Kohavi. Silicon Graphics, Inc., “Data Mining and Visualization.” 25 Sept. 1998. <http://ai.stanford.edu/~ronnyk/minesetHB.pdf#search=%22History%20of%20Mineset%22>.

[2] G.Conti, M. Ahamad, and J. Stasko, “Attacking Information Visualization System Usability Overloading and Deceiving the Human”, *SOUPS 2005*, Pittsburgh, PA, USA, July 6-8 2005.p.89-100.

[3] N. Gershon, S. Card, and S. Eick, “Information Visualization Tutorial”. *ACM-CHI99*, May 15-20 1999, p.149-150.

[4] G. Giuseppini, and M. Burnett, *Microsoft Log Parser Toolkit*, Syngress, Rockland, 2004.

[5] J. Heer, S. Card, and J. Landay, “Prefuse: A Toolkit for Interactive Information Visualization” *ACM-CHI 2005,* April 2-7, 2005, Portland Oregon, USA, p. 421-430.

[6] M. Heo, and S. Hirtle, “An Empirical Comparison of Visualization Tools to Assist Information Retrieval on the Web”, *Journal of the American Society for Information Science and Technology*, 52(8), p.666-675, 2001.

[7] H. Hochheiser and B. Shneiderman, “Using Interactive Visualizations of WWW Log Data to Characterize Access Patterns and Inform Site Design”, *Journal of the American Society for Information Science and Technology*, 52(4), p. 331-343, 2001.

[8] M. Kantardzic, J. Zurada, *Next Generation of Data-Mining Applications*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.

[9] K. Paviou, and R. Snodgrass, “Forensic Analysis of Database Tampering”, *ACM-SIGMOD’06*, June 27-29, 2006, p.109-120, 2006.

[10] R. Pilat, E. Valiati, and C. Freitas, “Experimental Study on Evaluation of Multidimensional Information Visualization Techniques”, *CLIHC’05*, October 23-26, 2005, p. 20-30.

[11] Purple Insight. Oct.1 2006. <http://www.purpleinsight.com/products/index/shtml>.

[12] T. Saracevic, “Information Science, *Journal of the American Society for Information Science,* 50(12), p. 1051-1063, 1999.

[13] J.Schauer, “Silicon Graphics, Powerful Solutions”. May 1998. <http://www.dmreview.com/article_sub.cfm?articleId=603>.

[14] A. Shen-Hsieh, and M. Schindler, “Data Visualization for Strategic Decision Making”, *American Institute of Graphic Arts Experience Case Study Archive*, p.1-17, 2002.

[15] “SGI Announces Agreement with Purple Insight for MineSet Data Mining and Real-Time 3D Visualization Software.” 2003. <http://www.sgi.com/company_info/newsroom/press_releases/2003/october/purple.html>.

[16] M. Souppaya, and K. Kent, *Guide to Computer Security Log Management (Draft)*, National Institute of Standards and Technology. Aithersburg, MD, 2006.

[17] N. Ye *The Handbook of Data Mining*, Lawrence Erlbaum Assoiates, New Jersey, 2003.

ITS: A DDoS Mitigating Architecture

Hikmat Farhat
Notre Dame University
Lebanon

Abstract- We propose a DDoS mitigation architecture that protects legitimate traffic from the large volume of malicious packets during a DDoS bandwidth attack. The system keeps a legitimacy list and gives higher priority to those packets that are on the list. The legitimacy list is kept up to date by keeping only the entries that complete the TCP three-way handshake and thus defeats IP spoofing. Entries in the list contain the IP address and the path signature of active TCP connections. A packet obtains high priority if its path signature strongly correlates with the corresponding path signature stored in the legitimacy list. We show that the scheme is efficient when deployed incrementally by using priority queuing at perimeter routers. An autonomous system (AS) can immediately benefit from our proposed system when deployed even if other ASs do not deploy it.

I. INTRODUCTION

While the open nature of the Internet is key to its success, it is also the main reason to its vulnerabilities. One of the most serious abuses of the Internet is the Denial of Service Attack (DoS) and its distributed version, the Distributed Denial of Service attacks (DDoS). Some of the DoS attacks received wide publicity, but, DoS attacks are in fact much more frequent [10]. Due to the relative simplicity of initiating them, bandwidth DoS attacks, in particular, have proven a difficult nut to crack. Today many services and critical infrastructure depend on the Internet, therefore protecting them from DoS attacks has become a crucial objective.

Protecting systems from bandwidth DDoS attacks has been an elusive goal. Detecting that a system or a network is under attack is in itself a difficult task. Even when we know that an attack is under way it is still a challenge to distinguish legitimate packets from malicious ones. While some DoS attacks use *spoofed* source IP addresses others use the legitimate address of compromised hosts, called *zombies*, to launch the attack. Because many attacks use IP spoofing, the determination of the source of the attack is an important step forward, albeit not enough. Toward this end many traceback schemes have been proposed [23,16,12,15]. A good number of DoS mitigation methods use a two-step approach. The first step consists of a learning phase (e.g. reconstruction of the attack paths) is needed to identify attack traffic signatures. In the second step filtering rules obtained from the "learning phase" are used to drop malicious traffic. The efficiency of the approach depends on the efficiency of the "learning" phase. Furthermore, traceback techniques usually take long time when the number of attackers is large and therefore cannot be used as a real-time response to attacks. In addition, the filters installed after the learning phase often block legitimate traffic as well leading to *collateral damage* which in itself is a form of DoS.

In a previous work [5] we have proposed the Implicit Token Scheme (ITS) to mitigate IP *spoofing*. The proposed method, while promising, had two shortcomings: it requires the estimate of the number of hops a packet has traveled, and could not be deployed incrementally, which is a serious shortcoming. We fix the above mentioned shortcomings in this paper by using a priority queuing at the perimeter routers of an ISP with packets having "better" signature matches given higher priority. This is made possible by the way the path signature is recorded. The same method obviates the need for the estimation of the number of hops.

The rest of the paper is organized as follows. Previous works on the DDoS problem are introduced in Section II. In Section III we present the objectives and assumptions of a DDoS mitigating architecture. Details of our proposed scheme are discussed in Section IV. The performed simulations and their results are shown in Section V. We conclude and provide pointer for future research in Section VI.

II. RELATED WORK

Most of the existing DDoS defense schemes are reactive in nature. The defense system becomes active when an attack is detected. Ideally, one would like to halt the attack and also determine the attacking host(s). One method, IP traceback, has focused on allowing the victim to trace the origin of malicious packets, which are usually spoofed [12,15,16,4,14]. The traceback methods require routers to stamp, with a certain probability, a mark in the IP header. When enough such packets have been collected the victim starts the process of reconstructing the path(s) that the attack packets have followed. While traceback schemes are important in finding the location of the attackers, they suffer from two shortcomings. First, the cost of the reconstruction algorithm becomes prohibitive when the number of attacking hosts is large. Second, most traceback approaches do not specify how (Ref [16] is a notable exception) to mitigate the DDoS attack once path reconstruction is completed. Many approaches have been proposed to solve the above-mentioned problems. Yaar et al. [23] have proposed a traceback algorithm that scale to thousand of attackers. Sung and Xu [16] use the concept of sub-channels to preferentially filter attack packets using the reconstructed paths.

Other approaches attempted to defend against DDoS attacks by filtering out IP packets with spoofed source addresses. One of the earliest such methods was the work by Ferguson et al. [6].

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 537–541.

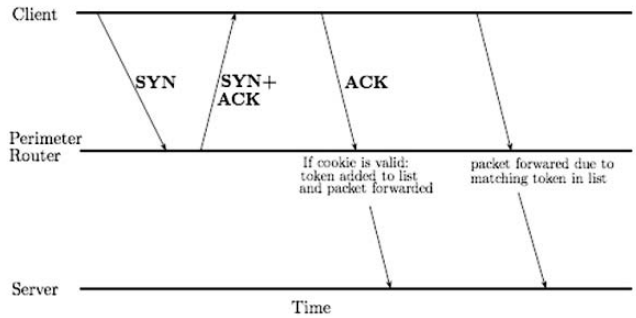


Fig. 1. Typical session of a legitimate client. Note that the SYN+ACK is received only by non-spoofed clients.

Their method requires the installation of ingress filtering at every ISP. Even if ingress filtering is universally deployed, an unrealistic assumption, IP addresses in the local network can still be spoofed. Another approach to ingress filtering is the SAVE protocol proposed by Li. et al. [9]. The distributed packet filtering method proposed by Park and Lee [11] discards spoofed IP packets using a route-based detection method even if only 20% of autonomous systems install such filters. Unfortunately their method requires the cooperation of thousands of autonomous systems. Jin [7] proposed a scheme where packets with spoofed addresses are identified by their hop count. The idea is to build a table in "peace time" that maps the source address of clients to the number of hops they need to reach the victim. During a DoS attack each packet is compared to the corresponding entry in the table. In practice, routes change every frequently, which makes the table entries obsolete very quickly.

The original idea for deterministic path identification is due to Yaar et al. [21]. They use the path identification which is a deterministic, as opposed to the probabilistic one used by most IP traceback methods, mark stamped by the intermediate routers on every packet as a way to distinguish malicious from legitimate users. Even if one assumes that the malicious signatures can be clearly identified the number of malicious and legitimate users having the same signature grows as the number of attackers grows, which quickly leads to self-inflicted DoS.

Anderson et al. [1] introduced the concept of capabilities whereby a sender first obtains a permission to send to the receiver from a Request-To-Send server (RTS) whose addresses are advertised by BGP as a community attribute. Yaar et al. [22] extended the idea where the sender obtains the permission explicitly from the receiver via a handshake protocol. Both method require routers to compute per packet hash functions and are vulnerable to attackers colluding with hosts co-located with the victim. Furthermore, the capabilities approach is vulnerable to the denial of capabilities attack [24]. A similar approach was proposed by Xu [20] to sustain the availability of Web servers and it uses HTTP redirect requests to prevent spoofed packets from reaching the victim.

III. ASSUMPTIONS AND DESIGN OBJECTIVES

Approaches based on building filters for malicious traffic will inevitably drop legitimate traffic along with illegitimate traffic, as the two cannot be totally distinguished from each other. ITS instead uses the opposite approach: it drops any traffic that has not proven itself to be genuine.

In building the Implicit Token Scheme we were guided by the following design objectives:

1. Require minimal or no changes to Internet protocols. Particular attention should be given to the feasibility of deployment.
2. The implemented method should achieve zero false positives otherwise it might lead to self-inflicted DoS.
3. The algorithms implemented by the routers for the common case should be simple and fast. This object is important for two reasons
   - Any change requiring significant per-packet computation by routers is unlikely to be accepted.
   - Intensive computation leads to slower router performance and might even lead to a DoS if the extra computation is slower than a table look up.

IV. THE ITS ARCHITECTURE

In the ITS architecture an ISP provides filtering service for its customers at its Point of Presence (POP). The filtering is done by maintaining a legitimacy list composed of *tokens* for all active TCP connections. An entry in the list is added only when a client completes the TCP handshake thus making sure that the entry is legitimate and up to date. Every packet carries a token, composed of the source IP address and path signature. The path signature is build by having intermediate routers stamp their mark in the identification field in the IP header. When a packet arrives at the perimeter router, its token is compared to tokens in the legitimacy list. The result of this comparison decides the fate of the packet.

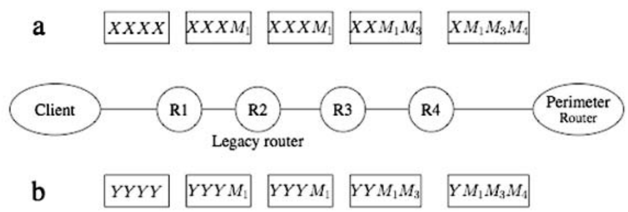


Fig. 2. Path signature in the presence of a legacy router with only 8 bits.
(a) packet that completes the TCP handshake: $XM_1M_3M_4$ is stored in the list.
(b) packet send at a later stage belonging to the same connection.

It is not practical to perform exact matches on path signatures for two reasons. First, if the path taken by a packet is not long enough, not all the bits in the identification field are overwritten and therefore two packets from the same client might have different signatures. Second, since any scheme has to be incrementally deployable, one would expect that not all the intermediate routers will stamp the packet, which also leads to the same result.

For all the above reasons the perimeter router assigns priority to packets related to how well the token carried by the packet matches values stored in the legitimacy list. After that the packet is added to the appropriate queue.

A. The Legitimacy List

The legitimacy list consists of tuples, where each tuple is composed of the source IP address and the corresponding path signature stored in the 16-bit IP identification field of the IP header. This field is marked by the routers along the path, from the source to the destination, where each router contributes 2 bits. When a packet with source IP address *s* arrives at the perimeter router it is assigned a priority depending on how well its path signature matches the signature stored in the legitimacy list for *s*. This way packets belonging to active TCP connections are given high priority. Building the legitimacy list, however, presents a problem.

The simplest approach would be to build the list during "normal traffic" and store the path signature for every IP address that connects to the target. This solution is not feasible for many reasons. First, the path signature may change from the time it is recorded to the time it is used as a result of routing changes. Second, during a spoofed attack the target sees a large number of previously unseen addresses, which leads to a large number of false positives [3]. Finally, a prospective target would require about a year to collect half of the Internet address space [3].

A better approach would be to dynamically add the path signature and the corresponding IP address of the client to the legitimacy list *after* the client completes the TCP three-way handshake. This way a *perimeter router* can allocate a large portion of the bandwidth, say 95%, to non-spoofed packets and the remaining 5% to rest. This method has none of the above-mentioned shortcomings. First, the path signature is up to date since it is the same one that is present during TCP connection establishment. Second, the legitimacy list will have the *exact* number of entries needed because any legitimate packet must arrive *after* the TCP handshake and therefore its corresponding entry will be present. Third, by delaying the addition of an entry until the TCP handshake is complete we are sure that the entry is correct because any attacker using a spoofed address will not be able to complete the TCP handshake.

While the above method protects all *established connections*, an attacker can still deplete the 5% of bandwidth allocated to connection establishment and denies legitimate users from establishing a new connection. To solve this problem we introduce a SYN cookie mechanism in the *perimeter routers*. A SYN cookie [17] is a special value of the Initial Segment Number (ISN) that allows a device to respond to a TCP connection request in a stateless manner. In this work it is used to allow a *perimeter router* to respond to TCP requests on behalf of the target. When a *perimeter router* receives a TCP SYN segment having destination IP address equal to that of the target it responds with a SYN+ACK on behalf of the target without maintaining any state information. Only when the third segment of the TCP handshake is received correctly, will the *perimeter router* forward the packet to the target. Note that this idea is not new and it has been already implemented in commercial products like the TCP intercept feature on CISCO devices [2]. A schematic of a typical session of a legitimate client is shown in Figure 1.

B. Path Signature

The path signature of a packet is a sequence of deterministic stamps marked by intermediate routers and stored in the 16-bit identification field of the IP header. Each router adds two bits, which represent a hash of the router IP number, to the identification field. When a packet with identification value *id* arrives at a router with mark *M,* the value of the identification field is modified as shown below:

$id=id << 2 +M$

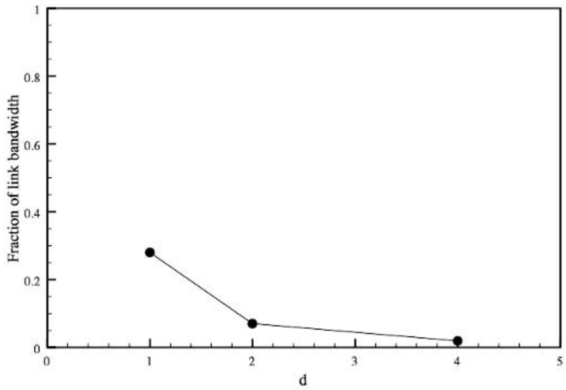


Fig. 3. Fraction of bandwidth consumed by spoofed packets for various values of d.

Where << is the bitwise left-shift operator. This way router stamps are always contiguous even in the presence of legacy routers. Since the IP identification field is 16-bits there are room for 8 stamps. Figure 2 illustrates this process by using an 8-bit field for simplicity.

Since every packet send by a client initially contains a different value in the identification field it is hard for the perimeter router to distinguish the original bits from the bits marked by intermediate routers. One possible solution is to estimate the number of hops a packet has traveled and use it to mask out the original bits. The idea is that most OSs use a well-known set of initial TTL values and by comparing the final TTL value one could estimate the number of hops. This method has two problems. First, it is not always possible to accurately determine the number of hops. More importantly, in the presence of legacy routers the number of hops that a packet has traveled is not equal to the number of router stamps, even if we could accurately determine the number of hops. Figure 2 illustrates this problem where for simplicity we have shown only 4 router marks. A client sends a packet with initial value of *XXXX* for the identification field and the signature is stored by the perimeter router as $XM_1M_3M_4$. At a later stage the same client sends a packet with initial value of *YYYY* and the value received by the perimeter router is $YM_1M_3M_4$, which does not match the stored value for that particular source IP address. The key idea is that even though the signatures do not match there are three contiguous router marks that match, $M_1M_3M_4$. This fact will be used in the next section to implement priority queuing.

C. Packet Filtering

In this section we assume that the packets arriving at the perimeter router are neither SYN packets nor a response to a SYN+ACK as these were dealt with in section IV.A using SYN cookies. The system maintains 9 different priority levels 0,…,8 where 0 is the lowest priority. When the perimeter router receives a packet it looks up the source IP address in the legitimacy list. If it does not exist the packet is given the lowest priority, 0. If it does exist, the number of *contiguous matches* between the packet signature and the one stored in the list is computed. This number of matches is used to assign the priority to the packet. As an example, consider Figure 2 again. Since the stored signature is $XM_1M_3M_4$ and the signature of the received packet is $YM_1M_3M_4$ then the number of matches is three and therefore the packet is given priority three.

It should be noted that the effectiveness of the system is due in large part to the fact that not all IP address space is valid as far as the perimeter router is concerned. Any packet with source IP not in the legitimacy list will automatically be assigned to the lowest priority queue no matter what its path signature is. An attacker packet with a random source IP address has the probability of 2^{-32} of matching the IP address of a legitimate client. Suppose that the system has 5000 active TCP connections at a certain period then the probability of an attacker packet with random source IP of matching a legitimate IP would be about one in a million. A smart attacker, however, might use the following approach.

If the attacker controls *n* distributed attacking hosts with IP's $ip_1,\ldots,ip_n$. She synchronizes their action as follows: ip_1 establishes a connection using its genuine IP address with the result that ip_1 is added to the legitimacy list with the corresponding path signature. Subsequently, all other attacking hosts send packets with source address ip_1. All those packets will match because ip_1 is already in the legitimacy list. Clearly the paths signature is needed in this case since not all attacking host will have the same path signature as ip_1.

We can use the above attacking strategy to show the effectiveness of our proposed method even if only the target AS deploys ITS. In this case there will be only one router stamp for every packet: the stamp of the ingress router of the target AS. Since the router stamp is 2-bits, then, given a source IP, the probability that the path signature also matches is 1/4. Thus using the synchronized attack strategy describe above only 25% of the attack packets will have the same path signature as ip_1. Of course if more ASs deploy ITS the scheme becomes much more effective and IP spoofing can be eliminated almost completely.

V. SIMULATION AND RESULTS

To test the efficiency of ITS, we performed a series of simulations with 100 "legitimate" hosts and 500 attacking hosts. The IP address and path signature of "legitimate" clients were recorded in the legitimacy list. Attacking hosts send data at the constant rate of 10M packets/s while the "legitimate" hosts send at the rate of 1M packets/s. The bandwidth of the link between the perimeter router and the target was set at 100 M packets/s. Therefore the data send by the legitimate clients saturates the target's link. The paths of both clients and attacker were selected randomly from CAIDA's skitter data [13]. Since we are interested in incremental deployment, only *d* intermediate routers stamped packets from sender to target with

This page intentionally blank

d=1,2,4. The *d* routers for each path were selected randomly with the condition that the border router of the target AS for that particular path is among them. The metric used to measure the performance of our method is the fraction of bandwidth of the link between the perimeter router and the target consumed by the attacking packets.

A simple priority queuing was performed where all high priority packets are processed before processing the low priority ones. For example, all priority 7 packets were forwarded before priority 6 and so on. While this is a simplistic model to adopt it serves as a good measure to gauge the effectiveness of the scheme. Two attack scenarios were considered.

In the first scenario the attacking hosts send packets with the source IP address randomly generated. Even when only the border routers of the target AS stamped packets, the attacking packets consumed less than 1% of the total bandwidth. A quick analysis will show us that this not a surprising result. The probability that a packet with random IP addresses matches the IP address of one of the 5000 clients is $5000x2^{-32}$. Therefore, an attacking host needs about 1000 seconds to get a single attacking packet to match a "legitimate" one.

In the second scenario, one of the attacking hosts, with source IP address ip_1, was assumed to have established a normal TCP connection to the target and therefore its source IP and path signature were added to the legitimacy list. The remaining attacking hosts used ip_1 as the source IP address. The results for *d=1,2,4* are shown in Figure 3. The results for *d=1* are very close to our analysis while the results for *d=2* and *d=4* show that the effectiveness of ITS in eliminating IP spoofing grows almost exponentially with *d*.

The simulation results have clearly shown that using ITS can largely eliminate IP spoofing even if not universally deployed. Furthermore, an ISP can immediately benefit from deploying ITS without waiting for other network providers to do so.

VI. CONCLUSION AND FUTURE WORK

We have proposed a priority queuing addition to ITS, a scheme to sustain the availability of TCP services even during large scale distributed denial-of-service attacks that involve tens of thousands of attackers. The proposed method gives preferential treatment to “legitimate” packets and delegates spoofed source IP address as well as packets bounced on unwitting *reflectors* to a low priority queue. IP spoofing and reflector attacks amplify the attackers resource. Therefore, ITS levels the playing field and forces potential attackers to marshal large resources, which is not always an easy task.

Even if the packets of an active TCP connection change their path in mid-flow they will not be dropped. They will be simply given low priority. Our method is completely transparent to clients and requires no change in the Internet protocols. Since the majority of traffic on the Internet is TCP this work offers a dependable solution to the ubiquitous DoS problem. The computational requirement on intermediate routers is minimal which increases the possibility of deployment and does not affect router performance even a GBPS speed.

We have also shown that the proposed architecture is incrementally deployable. When an ISP deploys ITS in its network it will immediately benefit from it by filtering out 25% of spoofed packets even if not other ISPs has deployed ITS.

REFERENCES

[1] T. Anderson, T. Roscoe, and D. Wetherall. Preventing internet denial of service with capabilities. *SIGCOMM Comput. Commun. Rev.*, 34(1):39–44, 2004.
[2] Configuring TCP Intercept, Cisco IOS Security Configuration Guide http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur c/.
[3] Michael Collins and Michael K. Reiter. An empirical analysis of targetresident DoS filters. In *IEEE Symp. on Secur. and Priv.*, 2004.
[4] D. Dean, M. Franklin, and A. Stubblefield. An algebraic approach to IP traceback. *ACM Trans. Inf. Syst. Secur.*, 5(2):119–137, 2002.
[5] Hikmat Farhat. Protecting TCP services from denial of service attacks. In *LSAD '06: Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense*, pages 155–160, 2006.
[6] P. Ferguson and D. Senie. Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing. *RFC 2827*, 2000.
[7] C. Jin, H. Wang, and K. G. Shin. Hop-count filtering: an effective defense against spoofed DDoS traffic. In *10th ACM Conference on Computer and Communications security*, pages 30–41, 2003.
[8] J. Jung, B. Krishnamurthy, and M. Rabinovich. Flash crowds and denial of service attacks: characterization and implications for cdns and web sites. In *11th international conference on World Wide Web*, pages 293–304, 2002.
[9] J. Li, J. Mirkovic, M. Wang, P. Reiher, and L. Zhang. Save: source address validity enforcement protocol. In *Proceedings of IEEE INFOCOMM, 2001*.
[10] D. Moore, G. Voelker, and S. Savage. Inferring internet denial of service activity. In *Proceedings of the 10th USENIX Security Symposium,* 2001.
[11] K. Park and H. Lee. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets. In *SIGCOMM '01:Proceedings of the 2001 SIGCOMM conference*, pages 15–26, 2001.
[12] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Network support for IP traceback. *IEEE/ACM Trans. Netw.*, 9(3):226–237, 2001.
[13] CAIDA's skitter initiative, http://www.caida.org.
[14] A. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer. Single-packet IP traceback. *IEEE/ACM Trans. Netw.*, 10(6):721–734, 2002.
[15] D. Song and A. Perrig. Advanced and authenticated marking schemes for IP traceback. In *Proceedings of IEEE INFOCOMM.*, 2001.
[16] M. Sung and J. Xu. IP traceback-based intelligent packet filtering: In *the 10th IEEE International Conference on Network Protocols*, 2002.
[17] D.J. Bernstein, http://cr.yp.com/syncookies.html.
[18] Haining Wang, Danlu Zhang, and Kang G. Shin. Change-point monitoring for the detection of DoS attacks. *IEEE Trans. Dep. Sec. Comput.*, 2004.
[19] Y. Xiang, Y. Lin, W. L. Lei, and S.J. Huang. Detecting DDoS attack based on network self-similarity. In *Communications,* 2004.
[20] J. Xu and W. Lee. Sustaining availability of web services under distributed denial of service attacks. *IEEE Trans. Comput.*, 52(2):195–208, 2003.
[21] A. Yaar, A. Perrig, and D. Song. Pi: a path identification mechanism to defend against DDoS attacks. In *IEEE Symp. on Secur. and Priv..*, 2003.
[22] A. Yaar, A. Perrig, and D. Song. SIFF: A stateless internet flow filter to mitigate DDoS flooding attacks. In *IEEE Symp. on Secur. and Priv.,* 2004.
[23] A. Yaar, A. Perrig, and D. Song. FIT: Fast Internet traceback. In Proceedings of *IEEE INFOCOMM,* 2005.
[24] K. Argyraki and D. Cheriton. Network Capabilities: The good, the bad and the ugly. In *HotNets-IV: The Fourth Workshop on Host Topics in Networks*, 2005.

Strong Designated Verifier Ring Signature Scheme

Ji-Seon Lee, Jik Hyun Chang
Dept. Computer Science, Sogang University,
1 Sinsu-dong, Mapo-gu, Seoul, Korea

Abstract-**In this paper, we propose a strong designated verifier ring signature scheme and discuss its security properties. The proposed scheme provides a way that leaks authoritative secrets to only a designated person anonymously by one of the group members and no one knows that the secret is from a group member or the recipient, except the recipient. This group is called a ring. We also propose a strong designated verifier ring signature with message recovery mechanism.**

Ⅰ. INTRODUCTION

In 2001, Rivest, Shamir, and Tauman [9] first formalized the concept of the ring signature. With a ring signature scheme, a signer can choose several members to form a temporary group called a ring and generate a ring signature without the assistance of the other ring members. Anyone can be convinced that the generated ring signature is from one of the ring members, but no one can identify the real signer among the ring members. This can be seen as a kind of non-interactive proof that a signer owns a witness of secret key that corresponds to one of n commitments of public keys. Subsequently, variant ring signature schemes have been proposed [1,2,3,4,5].

In 1996, Jakobsson, Sako, and Impagliazzo [6] introduced the concept of the designated verifier signature scheme which makes it possible for a signer to convince only the designated verifier that the signature is made by the signer. This is achieved since a designated verifier himself can efficiently simulate signatures that are indistinguishable from the signer's signature. Since the signer's public key and the designated verifier's public key are both included in the verification step, anyone can verify the signature. However, unlike ordinary digital signature schemes, no one can be convinced that who the real signer is, except the designated verifier. When the designated verifier Bob receives a signature from a signer Alice, he certainly trusts that it is from Alice upon verifying it, since he knows that he did not generate the signature himself. A designated verifier signature scheme is useful in some situations in which the signer should specify who may be convinced by the signer's signature. However, in some circumstances, the third party may be convinced with high probability that the signature intended for the designated verifier is actually generated by the signer. For example, the signature may be captured on the line by the third party before the designated verifier receives it. The third party can then confirm that the real signer is Alice. To protect the identity of the signer in such situations, the signer encrypts the signature with the designated verifier's public key so that only the designated verifier can get the signature generated by the signer with his secret key. This stronger requirement is called a strong designated verifier signature scheme and was discussed in [6]. Saeednia, Kremer, and Markowitch [10] proposed a new efficient designated verifier signature scheme which directly provides the strongness property without requiring any encryption of the signatures. In their scheme, the third party cannot even verify the signature since the secret key of the designated verifier is involved in the verification step. If the secret key of the designated verifier is exposed to the public, then anyone can verify the signature. However, still no one can confirm that the signature is from the signer or the designated verifier.

Rivest, Shamir, and Tauman [9] noticed that the designated verifier signatures can be implemented from ring signature scheme by including the verifier's public key in the ring. However, general ring signatures with simply involving the verifier's public key is not suitable to construct a strong designated verifier signature scheme.

Our Contribution: In this paper, we firstly propose a strong designated verifier ring signature scheme. Since the proposed scheme is a strong designated verifier signature, only the designated verifier can verify the signature and be convinced that the signature is made by one of the ring members. Since it is a ring signature, even the designated verifier does not have any idea who the real signer is among the n ring members. The proposed scheme would be useful in some situations. Suppose that someone wants to leak authoritative information only to a designated person or an institute in an anonymous way. He would sign that information which can be verified only by the designated recipient. The recipient knows that the information is from one of the ring members. However, except for the recipient, no one can tell from whom comes the information between a ring and a recipient since the recipient can simulate the signature in an indistinguishable way. As a variant of the proposed scheme, we also present a strong designated verifier ring signature scheme which prevents the third party from reading the message upon seeing the signature. We call this a strong designated verifier ring signature scheme with message recovery.

In section 2, we review Herranz and Saez's provably secure ring signature scheme. In section 3, we give some definitions of the proposed scheme and its security properties. In section 4, we propose our strong designated verifier ring signature and discuss its security properties. In section 5, we provide a strong designated verifier ring signature scheme which provides a message recovery mechanism. Some conclusions are made in section 6.

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 543–547.

Ⅱ. Herranz And Saez's Provably Secure Ring Signatures

Our scheme is based on Herranz and Saez's provably secure ring signature scheme [5]. In this section, we review their scheme.

Let p and q be two large primes such that $q \mid p-1$ and g be an element of $\mathbb{Z}_p$ of order q. The message to be signed is $m \in \mathbb{Z}_p$. We use (p,q,g) as common parameters to all n ring members and other participants in the scheme. Let H be a collision resistant hash function which outputs elements in $\mathbb{Z}_q$. Each member A_i, $1 \le i \le n$, has a private key $x_i \in \mathbb{Z}_q^*$ and the corresponding public key $y_i = g^{x_i} \bmod p$. Let L be a set of public keys of the ring members, that is, $L = \{y_1,...,y_n\}$.

Signature Generation. To generate a ring signature for a message m on behalf of n ring members $A_1,...,A_n$, a signer A_s, where $s \in \{1,...,n\}$, follows the below steps.

(1) For all $i \in \{1,...,n\}$, $i \ne s$, A_s randomly chooses $a_i \in \mathbb{Z}_q^*$ pairwise different and computes $R_i = g^{a_i} \bmod p$.

(2) A_s selects a random number $a \in \mathbb{Z}_q$.

(3) A_s computes $R_s = g^a \prod_{i \ne s} y_i^{-H(m,R_i)} \bmod p$. If $R_s = 1$ or $R_s = R_i$ for some $i \ne s$, then go to step (2).

(4) A_s computes $\sigma = a + \sum_{i \ne s} a_i + x_s H(m, R_s) \bmod q$.

(5) The signature is then $(L, m, R_1,..., R_n, h_1,..., h_n, \sigma)$, where $h_i = H(m, R_i)$, for all $1 \le i \le n$.

Signature Verification. The recipient checks that $h_i \stackrel{?}{=} H(m, R_i)$, for all $1 \le i \le n$. If this holds, the recipient verifies that the following equation holds or not.

$$g^{\sigma} = \prod_{1 \le i \le n} R_i \prod_{1 \le i \le n} y_i^{h_i} \bmod p$$

Herranz and Saez proved that their scheme satisfies anonymity and unforgeability in the random oracle model.

Ⅲ. Model

In this section, we define strong designated verifier ring signature scheme and its security properties.

Definition 1 A strong designated verifier ring signature scheme is defined by four procedures:

Key Generation Each member A_i, $1 \le i \le n$, in a ring has his key pair (x_i, y_i) and the designated verifier has his key pair (x_B, y_B).

Signature Generation If a user A_s, where $s \in \{1,...,n\}$, wants to compute a strong designated verifier ring signature on behalf of a ring that includes himself, he executes signature generation algorithm with input a message m, the public key y_B of the designated verifier, the public key list L of the ring members and the signer's secret key x_s.

Signature Verification The designated verifier checks the validity of the signature. The output of this algorithm is true or false.

Transcript Simulation The designated verifier simulates transcripts that are indistinguishable from the signatures generated by any of the ring members.

The proposed strong designated verifier ring signature scheme should satisfy the following requirements:

- Signer Anonymity for the Designated Verifier : The designated verifier cannot determine the real signer among the n ring members with probability greater than $1/n$.

- Signer Anonymity for the Third Party (Strong Designated Verifier Property) : Any verifier cannot determine the real signer among the n ring members and a designated verifier with probability greater than $1/(n+1)$.

- Unforgeability : Any attacker must not succeed in forging a valid transcript for some message m on behalf of a ring or a designated verifier without the secret key of any ring members or the designated verifier.

Ⅳ. Strong Designated Verifier Ring Signatures

In this section, we propose a strong designated verifier ring signature scheme. We also provide security properties of the proposed scheme.

A. The proposed scheme

As in section Ⅱ, we use (p,q,g) as common parameters to all participants. Each member A_i, $1 \le i \le n$, has a key pair (x_i, y_i), the designated verifier Bob has his key pair (x_B, y_B), and $L = \{y_1,...,y_n\}$.

Signature Generation. Among the n ring members, the signer A_s generates a strong designated verifier ring signature as follows :

(1) A_s randomly chooses $a_i \in \mathbb{Z}_q^*$ pairwise different and computes $R_i = g^{a_i} \bmod p$, for all $1 \le i \le n$, $i \ne s$.

(2) A_s chooses a random number $a \in \mathbb{Z}_q^*$.

(3) A_s computes $R_s = g^a \prod_{i \neq s} y_i^{-H(m,R_i)} \bmod p$. If $R_s = 1$ or $R_s = R_i$ for some $i \neq s$, then go to step (2).

(4) A_s selects a random number $k_1 \in \mathbb{Z}_q^*$ different from any of a_i s and computes $t = g^{k_1} \bmod p$.

(5) A_s selects a random number $k_2 \in \mathbb{Z}_q$ and computes $r = H(m, y_B{}^{k_2} \bmod p)$.

(6) A_s computes $K = a + \sum_{i \neq s} a_i + x_s H(m, R_s) \bmod q$.

(7) A_s computes $s = k_1^{-1}(Kr - k_2) \bmod q$.

(8) The signature is $(L, m, R_1, ..., R_n, h_1, ..., h_n, t, r, s)$, where $h_i = H(m, R_i)$, for all $1 \leq i \leq n$.

Signature Verification. The designated verifier checks whether $h_i = H(m, R_i)$ holds, for all $1 \leq i \leq n$. If this holds, the verifier computes A and checks the equality of the following formula:

$$A = \prod_{1 \leq i \leq n} R_i \prod_{1 \leq i \leq n} y_i^{h_i}$$

$$r \stackrel{?}{=} H(m, (A^r t^{-s})^{x_B}). \tag{1}$$

If (1) holds, the designated verifier accepts that the signature is from one of the ring members.

Transcript Simulation. The designated verifier Bob simulates a transcript for the message m with his private key in an indistinguishable way as follows:

(1) Bob randomly chooses $a_i' \in \mathbb{Z}_q^*$ pairwise different and computes $R_i' = g^{a_i'} \bmod p$, for all $1 \leq i \leq n$.

(2) Bob computes $A' = \prod_{1 \leq i \leq n} R_i' \prod_{1 \leq i \leq n} y_i^{H(m,R_i')}$.

(3) Bob selects a random number $t_1 \in \mathbb{Z}_q^*$ different from any of a_i' and computes $t' = (A')^{t_1^{-1}} \bmod p$.

(4) Bob selects a random number $t_2 \in \mathbb{Z}_q$ and computes $r' = H(m, (A')^{x_B t_1^{-1} t_2} \bmod p)$.

(5) Bob computes $s' = t_1 r' - t_2 \bmod q$.

(6) The signature is $(L, m, R_1', ..., R_n', h_1', ..., h_n', t', r', s')$, where $h_i' = H(m, R_i')$, for all $1 \leq i \leq n$.

B. Analysis

In this section, at first we show that the proposed scheme is correct.

Correctness The signature generated by A_s and the signature simulated by Bob can be verified correctly by the verification step.

1. The signature by one of the ring members can be verified correctly by Bob as follows:

$$\begin{aligned} r &= H(m, y_B{}^{k_2} \bmod p) \\ &= H(m, (g^{Kr - sk_1})^{x_B}) \\ &= H(m, \{g^{Kr}(g^{k_1})^{-s}\}^{x_B}) \\ &= H(m, \{(\prod_{1 \leq i \leq n} R_i \prod_{1 \leq i \leq n} y_i^{H(m,R_i)})^r t^{-s}\}^{x_B}). \end{aligned}$$

2. If Bob reveals his secret key to the public, the simulated signature can be verified correctly by any third party as follows.

$$\begin{aligned} & H(m, \{(\prod_{1 \leq i \leq n} R_i' \prod_{1 \leq i \leq n} y_i^{H(m,R_i')})^{r'} t'^{-s'}\}^{x_B}) \\ &= H(m, \{(\prod_{1 \leq i \leq n} R_i' \prod_{1 \leq i \leq n} y_i^{H(m,R_i')})^{r'} (A')^{t_1^{-1}(t_2 - t_1 r')}\}^{x_B}) \\ &= H(m, (\prod_{1 \leq i \leq n} R_i' \prod_{1 \leq i \leq n} y_i^{H(m,R_i')})^{x_B t_1^{-1} t_2}) \\ &= r' \end{aligned}$$

Now we show that the proposed scheme satisfies security requirements stated in the previous section.

Signer Anonymity for the Designated Verifier : In order to prove that the signature is anonymous to the designated verifier , we show that the designated verifier has probability $1/n$ to guess which member of the ring actually generates a given signature. With similar approach to the proof of signer anonymity in the ring signature scheme [5], the probability that one of the ring member generates a signature correctly is $1/\{q(q-1)(q-2)...(q-n)(q-n-1)\}$. Since any member of the ring has the same probability to generate a signature, the proposed scheme preserves the anonymity property.

Signer Anonymity for the Third Party (Strong Designated Verifier Property): To prove that the proposed scheme is a designated verifier ring signature scheme, we show that the simulated transcripts by Bob are indistinguishable from the transcripts generated by any of the ring members. Since the simulated signature depends on the random values of $t_1 \in \mathbb{Z}_q^*$, $t_2 \in \mathbb{Z}_q$, and a_i , for all $1 \leq i \leq n$, the probability that the transcript simulated by Bob correctly is $1/\{q(q-1)(q-2)...(q-n)(q-n-1)\}$. This probability is the same as the probability that a signature is generated by a ring member discussed above. Therefore, the probability that the third party can guess the real signer among $n+1$ participants – n ring members and the designated verifier – is $1/(n+1)$.

Since Bob has the ability to simulate the transcript in an indistinguishable way, no one can tell that the signature is from any of the ring members or Bob. The proposed scheme satisfies the strongness property of a designated verifier signature scheme since Bob's secret key is included in the signature verification step. That is, only Bob can verify the signature. If Bob's secret key is compromised, then anyone can verify the signature. However, still no one can tell that the signature is from a ring or Bob.

Unforgeability : While the signature should be simulated by Bob, it should not be forged or simulated by any third party. We can think of two scenarios that the attacker could try out to forge a signature.

Scenario 1. The attacker would try to generate a signature as any of the ring members does. The attacker follows the steps (1) through (5) in the signature generation in section 4. Next the attacker tries to compute K followed by s which should satisfy the verification step. However, to do this, the attacker should have any of the ring members' secret key which means that he should solve the discrete logarithm problem.

Scenario 2. The attacker would try to simulate a signature as Bob does. The attacker follows the steps (1) and (3) in the transcript simulation in section 4. He then tries to compute r' followed by s' which should satisfy the verification step. Likewise in scenario 1, to do this, the attacker should have the designated verifier's secret key which means that he should solve the discrete logarithm problem.

In both scenarios, the successful forgery by any third party means that the attacker solves the discrete logarithm problem.

Ⅴ. Strong Designated Verifier Ring Signature Scheme With Message Recovery

Nyberg and Ruepple [7,8] first proposed a signature scheme which has message recovery mechanism. In their scheme, instead of a message being attached to a signature, a message is recovered and a signature is verified by the recipient at the same time. In this section, with some modification we propose a strong designated verifier ring signature scheme with message recovery. This would be useful in some situations in which upon seeing a signature, the third party can neither identify the real signer nor get the original message. For signature generation, A_s follows the same steps in the signature generation in the proposed strong designated verifier ring signature, except step (5). In step (5), to add message recovery mechanism, the signer selects a random number $k_2 \in \mathbb{Z}_q$ and then computes $c = m{y_B}^{-k_2} \bmod p$ and $r = H(m, g^{k_2} \bmod p)$. The signature is then $(L, R_1, ..., R_n, h_1, ..., h_n, t, c, r, s)$. When Bob receives this transcript, Bob verifies whether $h_i = H(m, R_i)$ holds for all $1 \le i \le n$. If this holds, the verifier computes $A = \prod_{1\le i\le n} R_i \prod_{1\le i\le n} {y_i}^{h_i}$ and recovers the message m by $m = c(A^r t^{-s})^{x_B} \bmod p$. With this recovered message m, Bob verifies the equation $r = H(m, (A^r t^{-s}))$ holds or not. If this holds, the designated verifier accepts that the signature is from one of ring members.

Bob can also simulate the signature with message recovery mechanism as the signature simulation in section 4. That is, Bob follows the same steps (1) through (3) in signature simulation, and then computes $c' = m'(A')^{-t_1^{-1} t_2 x_B} \bmod p$ and $r' = H(m', (A')^{t_1^{-1} t_2} \bmod p)$ in step (4). The step (5) is the same. The simulated signature is $(L, R_1', ..., R_n', h_1', ..., h_n', t', c', r', s')$ which is satisfied by the message recovery and the signature verification step as follows:

$$A' = \prod_{1\le i\le n} R_i' \prod_{1\le i\le n} {y_i}^{h_i}$$

$$\begin{aligned} m &= c'((A')^{r'} t'^{-s'})^{x_B} \\ &= c'((A')^{r'} (A')^{t_1^{-1}(t_2 - t_1 r')})^{x_B} \\ &= c'((A')^{t_1^{-1} t_2 x_B}) \\ r' &= H(m, (A')^{r'} t'^{-s'}) \\ &= H(m, (A')^{r'} (A')^{t_1^{-1}(t_2 - t_1 r')}) \\ &= H(m, (A')^{t_1^{-1} t_2}). \end{aligned}$$

Confidentiality : Since this scheme provides a message recovery mechanism, it should provide confidentiality in addition to other security requirements. Confidentiality means that except the recipient Bob, anyone else cannot extract the message m from the signature $(L, R_1, ..., R_n, h_1, ..., h_n, t, c, r, s)$. In the proposed scheme, since Bob's secret key is required to recover the message m from the signature, confidentiality is provided.

Ⅵ. Conclusions

In this paper we propose a strong designated verifier ring signature scheme which provides a way to leak authoritative secrets anonymously by one of the ring members and no one knows that the secret is from a ring member or a designated verifier, except the designated verifier. Our scheme guarantees that only the designated verifier Bob can be convinced that the signature is really from one of the ring members. Even if Bob's secret key is exposed to the public, no one can tell that the signature is from one of the ring members or Bob. Furthermore, since it is a ring signature, even Bob does not know who the real signer is among the n ring members. We show that our scheme provides signer anonymity, unforgeability, and the strong designated verifier property and also propose a strong designated verifier ring signature scheme which has message recovery mechanism.

References

[1] M. Abe, M. Ohkubo, and K. Suzuki, "1-out-of-*n* signatures from a variety of keys," *Advances in Cryptology – ASIACRYPT 2002, LNCS 2501*, pp. 415-432, 2001.

[2] E. Bresson, J. Stern, and M. Szydlo, "Threshold ring signatures and applications to ad-hoc groups," *Advances in Cryptology – CRYPTO 2002, LNCS 2442*, pp. 465-480, 2002.

[3] S. Chow, S. Yiu, and L. Hui, "Efficient identity based ring signature," *Applied Cryptography and Network Security, Third International Conference, ACNS 2005, Proceedings, LNCS 3531*, pp.499-512, 2005.

[4] S. Chow, K. Liu, K. Wei, and T. Yuen, “Ring signatures without random oracles,” *Proceedings of AsicCCS 2006*, pp. 297-302, 2006.

[5] J. Herranz, and G. Saez, “Forking lemmas for ring signature schemes,” *Indocrypt'03, LNCS 1403*, pp. 406-421. 2003.

[6] M. Jakobsson, K. Sako, and R. Impagliazzo, “Designated verifier proofs and their applications,” *Advances in Cryptology - EUROCRYPT '96, LNCS 1070,* pp. 143-154, 1996.

[7] K. Nyberg, and R.A. Rueppel, “A new signature scheme based on the DSA giving message recovery,” *Proceedings of the First ACM Conference on Computer and Communication Security*, pp. 58-61, 1993.

[8] K. Nyberg, and R.A. Rueppel, “Message recovery for signature schemes based on the discrete logarithm problem,” *Advances in Cryptology - EUROCRYPT '94. LNCS 950*, pp. 182-193, 1994

[9] R.L. Rivest, A. Shamir, and Y. Tauman, “How to leak a secret,” *Advances in Cryptology – ASIACRYPT 2001, LNCS 2248*, pp.257-265, 2001.

[10] S. Saeednia, S. Kremer, and O. Markowitch, “An effcient strong designated verifier signature scheme,” *Information Security and Cryptology - ICISC'03, LNCS 2971*, pp. 40-54, 2003.

Performance of Enhanced Distance Vector Multipath Routing

Kanwalinder Jit Kaur
Lecturer
Department of Computer Science, Punjabi University, Patiala, India
Tel.:+91-9815612289(Cell), +91-175-2286224(R), +91-175-3046312(O)
Fax: +91-175-3046313
E-mail: kanwal_gheek1@yahoo.com
Dr. Jyotsna Sengupta
Reader
Department of Computer Science, Punjabi University, Patiala, India
Tel.:+91-9316201605(Cell), +91-175-2215251(R), +91-175-3046312(O)
Fax: +91-175-3046313
E-mail: jsengupta1@lycos.com

Abstract-In this paper, a new algorithm named Enhanced Distance Vector routing algorithm has been developed to make the existing distance vector routing secure. This algorithm has been extended to multipath routing to make multipath data transmission secure. The performance of this algorithm has been found out by using simulation environment of *ns-2*. The results show that multipath Enhanced Distance Vector performs better than single path Enhanced Distance Vector in terms of throughput by 10 times better and cumulative distribution better by about 46%.

1. INTRODUCTION

Distance vector routing algorithms operate by having each router maintain a table, that is, a vector, giving the best known distance to each destination and which path to use to get there [7]. These tables are updated by exchanging information with the neighbors. Each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. These entries are preferred outgoing line to use for that destination, and an estimate of the time or distance to that destination. The metric used might be number of hops, time delay, and total number of packets queued along the path.

For enhancing the security, the most obvious solution is to use a complex encrypting algorithm. It is common for present day attackers to check a traffic stream of a desired IP source/destination pair and record the transaction data, then apply sophisticated tools to decode the information. Although this is a time consuming complex operation, with the right deciphering software, there is a possibility that the file might be decodable, thus releasing confidential/secured information [3]. Hence, the use of a complex encryption tool alone will not be the complete solution. Therefore, coupling the encryption process with a multipath routing topology will increase the security of data. Traditional protocols like the link state protocol [2] and the shortest path protocol [2] will provide data communication through the shortest path, which commonly is a fixed single path that does not change, considering the stability of the network connections and switches/routers. If the path from source to destination is monitored, there are many intermediate nodes. An attacker can try to hack into any one of these intermediate nodes. But in the proposed topology of multipath Enhanced Distance Vector, the path is never fixed and hence this becomes an effective way of protecting the network from attackers. Even if an attacker gains access and copies the data at a particular node, it will not be of significance because only a part of the encrypted message is obtained.

Ad hoc On Demand Distance Vector Routing (AODV) [6] algorithm has been proposed for mobile nodes. These nodes do not require intervention of any central point or existing infrastructure. Ad hoc on-demand multipath distance vector (AOMDV) [5] algorithm is a multipath extension to ad hoc on-demand distance vector (AODV), which is a single path routing protocol. AOMDV computes multiple loop-free and link-disjoint paths. In mobile ad hoc networks, one might expect multipath routing to provide some robustness to link failures and facilitate the transmission of packets along paths that avoid regions of congestion. TCP has been considered as goodput as the metric of performance to improve network performance in terms of the achieved throughput [9]. The performance issues of destination-sequenced distance vector (DSDV) and ad-hoc on-demand distance vector (AODV) routing protocols for mobile ad hoc networks have been measured [4]. The issue has been discussed for mobile hosts but more or less the same has not been discussed for static hosts, where performance of multipath algorithm is out performing than that of single path.

II. ALGORITHM DEVELOPED

Enhanced Distance Vector (EDV) routing algorithm has been implemented in two ways, single path and multipath. The multiple paths chosen for data transmission are of equal

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 549–552.

cost. Analysis has been done on the working and efficiency of the algorithm on these two different scenarios.

Enhanced Distance Vector Routing is the implementation of Distributed Bellman Ford (or Distance Vector) routing. The implementation sends periodic route updates every *advertInterval*. This variable is a class variable. Its default value is 2 seconds. In addition to periodic updates, each agent also sends triggered updates, it does this whenever the forwarding tables in the node change. This occurs either due to changes in the topology, or because an agent at the node received a route update, and recomputed and installed new routes. Each agent employs the split horizon with poisoned reverse mechanisms to advertise its routes to adjacent peers. "Split horizon" is the mechanism by which an agent will not advertise the routes to a destination out of the interface that it is using to reach that destination. In a "Split horizon with poisoned reverse" mechanism, the agent will advertise that route out of that interface with a metric of infinity. Each DV agent uses a default preference_ of 120. The value is determined by the class variable of the same name. Each agent uses the class variable Infinity (set at 32) to determine the validity of a route. Because distance vector routing works in theory but has a serious drawback in practice that it converges to the correct path, it may do that slowly. Moreover, it reacts rapidly to good news, but leisurely to bad news. To overcome this, count to infinity, problem variable Infinity has been set.

III. SIMULATION ENVIRONMENT

NS-2 [1, 8] has been used for simulation study. The hosts are placed on a square field of 1000m x 1000m. The constant bit rate (CBR) traffic is used in the simulation. Each connection is specified as a randomly chosen source-destination (S-D) pair. The packet size is fixed as 512 bytes. The packet sending rate is 4 packets per second. Each connection starts at a time randomly chosen from 0 to 100 seconds. Simulations are run for 8 simulated seconds. Each data point represents an average of ten runs with identical traffic model.

IV. PERFORMANCE ANALYSIS

The performance of the algorithms has been evaluated using ns-2 simulator. The following key issues have been addressed:

1. Throughput of generating packets.
2. Throughput of sending bits vs minimal simulation End to End delays.
3. Throughput of sending bits vs average simulation End to End delays.
4. End to End Simulation Delays vs Cumulative Distribution.

Results:

1. Throughput of generating packets: The number of packets generated in multipath is 10 times higher than number of packets generated in single path with in same simulation time, where packet generating pattern is almost the same.

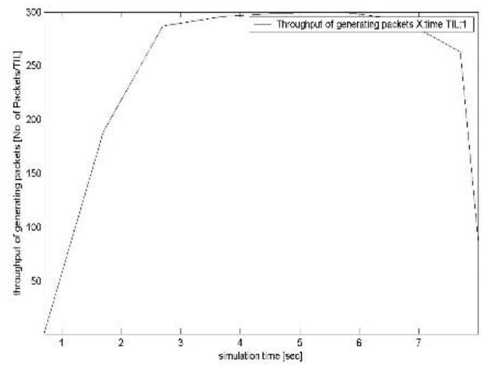


Fig.1 Throughput of generated packets for Single Path Enhanced Distance Vector

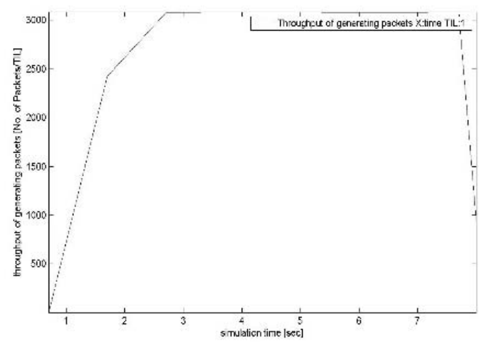


Fig.2 Throughput of generated packets for Multipath Enhanced Distance Vector

2. Throughput of sending bits vs minimal simulation End to End delays: In multipath throughput of sending bits with respect to minimal simulation End to End delay shows 10 times better results than single path.

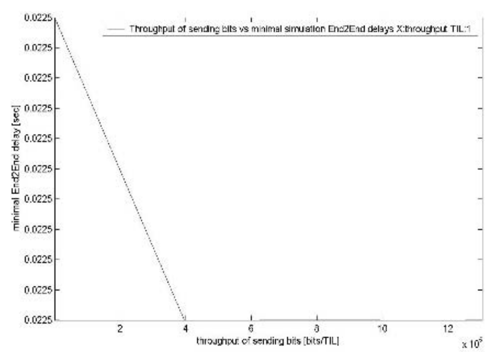


Fig. 3 Throughput of sending bits vs minimal simulation End to End delays for Single Path Enhanced Distance Vector

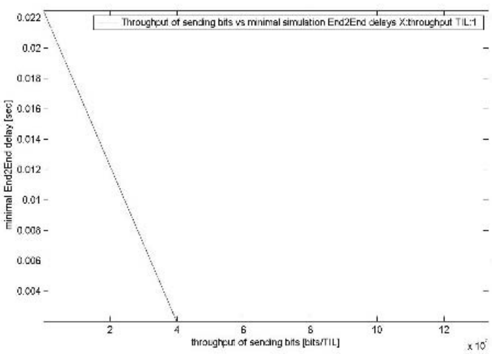


Fig. 4 Throughput of sending bits vs minimal simulation End to End delays for MultiPath Enhanced Distance Vector

3. Throughput of sending bits vs average simulation End to End delays: Throughput of sending bits in multipath is about 10 times more than single path. Whereas pattern is not the same.

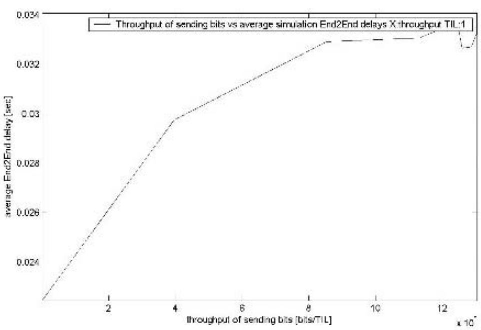


Fig. 5 Throughput of sending bits vs average simulation End to End delays for Single Path Enhanced Distance Vector

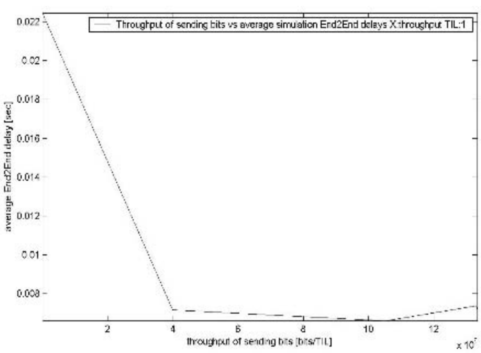


Fig. 6 Throughput of sending bits vs average simulation End to End delays for MultiPath Enhanced Distance Vector

4. End to End Simulation Delays vs Cumulative Distribution**:** Cumulative Distribution is high for multipath in comparison to single path. This shows that about 46% more number of packets is distributed in less time in multipath in comparison to single path.

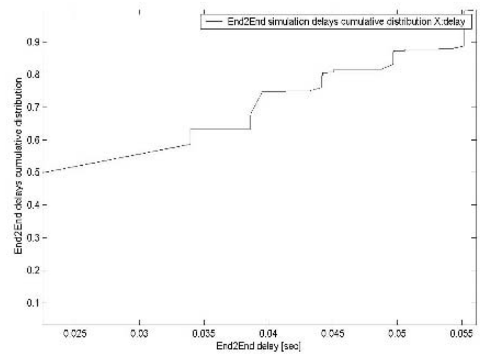


Fig. 7 End to End Simulation Delays vs Cumulative Distribution for Single Path Enhanced Distance Vector

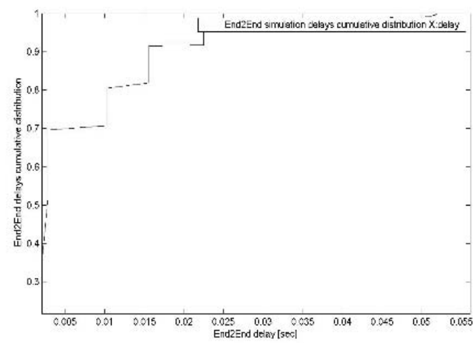


Fig. 8 End to End Simulation Delays vs Cumulative Distribution for Multipath Enhanced Distance Vector

V. CONCLUSION AND FUTURE WORK

From above given facts, we conclude that Multipath Enhanced Distance Vector performed better than Single path Enhanced Distance Vector, as Throughput of generating packets for multipath is ten times higher than single path. Throughput of sending bits with respect to average and minimal simulation end to end delay is about 10 times better for multipath than single path. Cumulative distribution of delivered packets is about 46% high with multipath than single path. Moreover messages could be made more secure from hackers using multipath and congestion could be reduced. Performance evaluation of Enhanced Distance Vector has been done on static nodes and static links. It could be extended for dynamic nodes (mobile) and dynamic links (link going up and down).

VI. REFERENCES

[1] Fall K. and Varadhan K., ns notes and documentation, 1999, available from http://www-mash.cs.berkeley.edu/ns/.

[2] Gouda Mohamed G., “Elements of Network Protocol Design”, A Wiley-Interscience Publication, John Wiley & Sons, Inc., New York, 1998.

[3] Jie Yang, “Improving Network Security by Multipath Traffic Dispersion”, Military Communications Conference, MILCOM’01. Communications for Network-Centric Operations: Creating the Information Force, Volume 1, pp. 34-38, 28-31 Oct. 2001.

[4] Lu Yi, Wang Weichao, Zhong Yuhui, Bhargava Bharat, “Study of Distance Vector Routing Protocols for Mobile Ad Hoc Networks”, the proceedings of First IEEE International Conference on Pervasive Computing and Communications, (PerCom 2003), pp. 187-194, 23-26 March 2003.

[5] Marina Mahesh K., Das Samir R., “On-demand Multipath Distance Vector Routing in Ad Hoc Networks”, the proceedings of Ninth IEEE International Conference on Network Protocols, pp. 14-23, 11-14 Nov. 2001.

[6] Perkins Charles E., Royer Elizabeth M., **“**Ad-hoc on-demand distance vector routing**”,** the proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications, WMCSA '99, pp. 90-100, 25-26 Feb. 1999.

[7] Pham Peter P. and Perreau Sylvie, “Performance Analysis of Reactive Shortest Path and Multipath Routing Mechanism with Load Balance”, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2003. Volume 1, pp. 251 – 259, 30 March-3 April 2003.

[8] http://www.isi.edu/nsnam/ns/doc

[9] Ye Zhenqiang, Krishnamurthy S.V, Tripathi S.K, “Effects of multipath routing on TCP performance in ad hoc networks”, IEEE Global Telecommunications Conference, GLOBECOM '04, Volume 6, pp. 4125-4131, 29 Nov.-3 Dec. 2004.

Coded 4-PAM OFDM for High Rate Data Links

M. A. Nasr*, Heba A. Shaban† and M. A. Mangoud‡
*Computer Engineering Department and ‡ Electronics and Communications Department
Arab Academy for Science and Technology (AAST), Alexandria, Egypt
*E-mail: mnasr@vt.edu, and ‡ E-mail: mohab@vt.edu
†Virginia Tech. VT-MENA program, AAST, Alexandria, Egypt
E-mail: hshaban@vt.edu

***Abstract*- Orthogonal Frequency Division Multiplexing (OFDM) is one of the best solutions for wideband communication applications. M-ary PAM – OFDM achieves comparable power and bandwidth efficiencies with less system complexity to ordinary OFDM systems that use M-ary QAM or M-ary PSK. Basically, OFDM is sensitive to Carrier Frequency Offset (CFO) and to phase noise. Performance evaluation of M-ary PAM OFDM over AWGN impaired with CFO and white phase noise is investigated in this paper with simulation. Also, the coded 4-PAM OFDM is proposed and its performance is evaluated over AWGN impaired with CFO and phase noise with simulation in comparison with un-coded 4-PAM OFDM.**

I. Introduction

Modern Communication systems employ Orthogonal Frequency Division Multiplexing (OFDM) due to its high spectral efficiency. In traditional wire line and PCM communications Pulse Amplitude Modulation (PAM) was widely used. However, higher data rates are required in emerging communication applications. The proposal of M-ary PAM OFDM is introduced in [1]. M-ary PAM-OFDM achieves comparable power and bandwidth efficiencies compared to ordinary OFDM systems that use M-ary QAM and M-ary PSK. Furthermore, employing M-ary PAM OFDM requires less computational burden and less system complexity compared to ordinary OFDM systems. Basically, M-ary PAM OFDM achieves bandwidth saving on the expense of losing power efficiency; resulting in a high Peak-Average Power Ratio (PAPR) [1]. One way to reduce PAPR is to introduce coding [2]. Coding also serves for increasing capacity; supporting the increasing demand for higher data rate communications.

OFDM is sensitive to Carrier Frequency Offset (CFO) and to phase noise, which degrade the system performance because of the loss of sub-carrier orthogonality, as well as the introduction of Inter-Carrier Interference (ICI). The effect of CFO on OFDM has been studied in [2]. Also, the effect of phase noise on OFDM has been studied in [3].

In this paper the performance of M-ary PAM OFDM over AWGN channel impaired with CFO and white phase noise is evaluated. The proposed coded 4-PAM OFDM is introduced and its performance over AWGN channel with phase noise and CFO compared to un-coded 4-PAM OFDM is studied.

The organization of this paper is as follows. A brief overview of M-ary PAM OFDM is presented in section II. White phase noise model and performance evaluation using simulation of M-ary PAM over AWGN channel impaired with white phase noise are given in section III. Moreover, performance evaluation of M-ary PAM OFDM over AWGN channel with CFO is evaluated via simulation in section IV. In section V proposed coded 4-PAM OFDM is introduced and its performance is compared to un-coded 4-PAM OFDM over AWGN channel with white phase noise and CFO with simulation. Finally, conclusions are discussed in section VI.

II. Overview of M-ary PAM OFDM

M-ary QAM-OFDM or M-ary PSK OFDM, either modulation schemes when applied on OFDM systems, the minimum frequency separation that guarantees orthogonal sub-carriers is $1/T$, where T is the symbol duration. However, if the sub-carriers differ only in their frequencies and amplitudes and their phases are the same, then the minimum required frequency spacing is $(1/2T)$; as shown in Fig.1. ; resulting in a less complex system. Also, M-ary PAM OFDM can be implemented using Discrete Cosine Transform (DCT), which is an efficient algorithm and can be implemented using real additions [1].

M-ary PAM OFDM as defined in [1]

$$s(t)=\sum_{k=0}^{N-1} A_k \cos(2\pi \frac{k}{2T} t) \qquad (1)$$

M-ary PAM OFDM exhibits a bandwidth saving ratio compared to M-QAM and M-PSK OFDM given by [1]

$$\text{Bandwidth Savings} = \frac{2(N+1)}{N+3} \qquad (2)$$

However, there is a tradeoff between bandwidth saving and power efficiency. PAPR compared to M-ary PSK OFDM [1]

$$\mathrm{P}APR_{\deg r(PO)} = \frac{3(M-1)}{M+1} \qquad (3)$$

The number of required samples for M-ary PAM OFDM is half that required for either M-ary QAM or M-ary PSK OFDM systems. Moreover, this results in a reduced system complexity [1], [5].

1 VT-MENA program: Virginia Tech -Middle East and North Africa program

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 553–556.

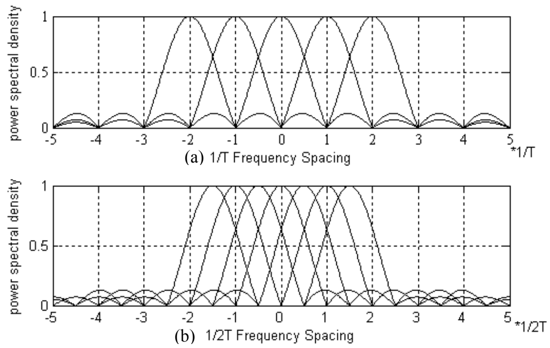


Fig. 1. Spectra of OFDM subcarriers (a) QAM or PSK OFDM, the subcarrier separation is $1/T$. (b) MASK OFDM, the subcarrier separation is $1/2T$ [3]

III. Performance Evaluation of M-ary PAM OFDM Over AWGN Channel with White Phase Noise

Phase noise is modeled as a phase modulation of the carrier for simulation purposes [4]. The presence of phase noise degrades the performance of OFDM systems causing ICI [4].

The effect of phase noise $\varphi(n)$ on the received signal $r(n)$ in terms of the transmitted signal with AWGN $x(n)$ as in [4]:

$$r(n) = x(n).e^{j\varphi(n)} \tag{4}$$

As phase noise variance increases, the degradation in system performance becomes more severe [4]. Simulation model of OFDM system is shown in Fig. 2. System BER performance evaluation via simulation of the effect of white phase noise on M-ary PAM OFDM was performed and the results are shown in Fig.3 for the case of no-phase noise and for the case of the presence of phase noise with noise variances σ^2= 0.02 rad^2 and for σ^2= 0.07 rad^2 as shown in Fig.4.

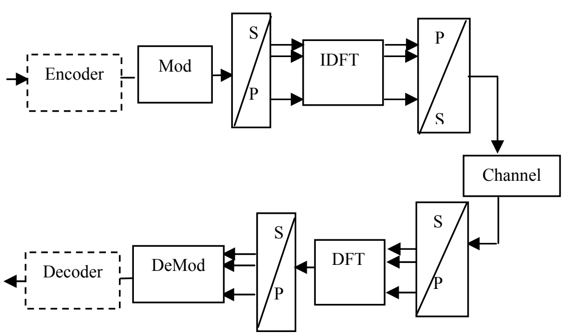


Fig. 2. Simulation model of OFDM transceiver (128 carriers) over AWGN channel [2]

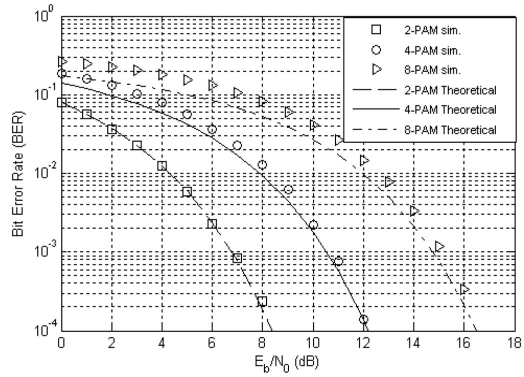


Fig. 3. BER of M-ary PAM OFDM (128 carriers) over AWGN channel theoretically and via simulation

Signal to noise ratio degradation due to phase noise [4]

$$\deg r_{SNR} = 10.\log(1+\sigma^2 \frac{E_s}{N_0}) \text{ dB} \tag{5}$$

It can be shown from Fig.3 that BER performance degrades as phase noise variance increases. Substituting in (5), for the case of 2-PAM OFDM at signal to noise ratio of 6dB and phase noise variance of 0.02 rad^2 signal to noise ratio degradation is equal to 0.49dB, and for noise variance of 0.02 rad^2 degradation is equal to 1.52dB for the same BER. Simulation results verify these values.

IV. Performance Evaluation of M-ary PAM OFDM Over AWGN Channel with CFO

OFDM systems are sensitive to CFO, and for a large number of sub-carriers inter-carrier orthogonality is destroyed and ICI is introduced. Consequently, the system performance degrades [3].

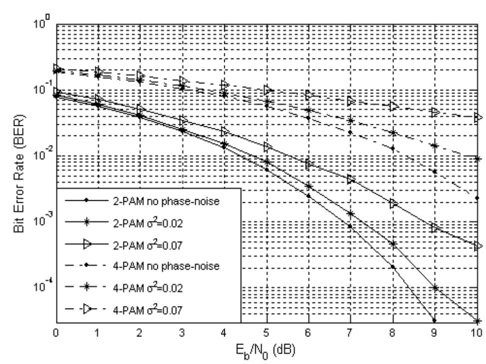


Fig. 4. BER of M-ary PAM OFDM (128 carriers) over AWGN channel impaired with phase noise

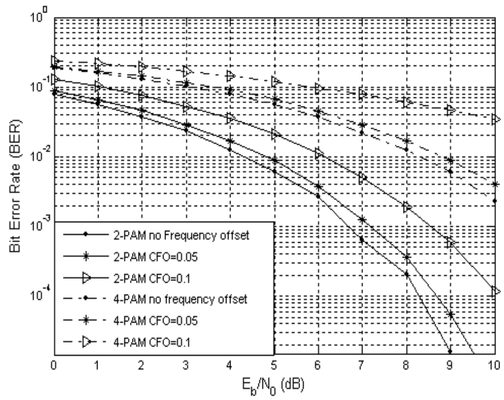


Fig. 5. BER of M-ary PAM OFDM (128 carriers) over AWGN channel impaired with CFO

ICI coefficients (S_k) as defined in [2]:

$$S_k = \frac{\sin \pi(k+\varepsilon)}{N \sin \frac{\pi}{N}(k+\varepsilon)} \exp[j\pi(1-\frac{1}{N})(k+\varepsilon)] \tag{6}$$

Where, ε is the normalized frequency offset which is the ratio between the CFO and the adjacent sub-carrier spacing [3].

The effect of CFO on M-ary PAM OFDM is evaluated via simulation over AWGN as depicted in Fig.5 for values of CFO= 0.05 and CFO= 0.1 respectively. Notably, as CFO increases, system BER performance degrades for the same modulation scheme order. These results are verified for the cases of 2-PAM OFDM and 4-PAM OFDM.

V. Proposed Coded 4-PAM OFDM over AWGN Channel Impaired with CFO and White Phase noise

Basically, for high speed data links employing PAM, the 4-PAM was specifically addressed in [6] instead of 2-PAM to counteract the channel attenuation. Also, 4-PAM was introduced with convolutional coding in [7] for high speed inter-chip interconnects that require faster links.

M-ary PAM OFDM systems exhibit high PAPR, as power efficiency is traded for bandwidth efficiency [3]. One method to reduce PAPR is to introduce coding. There have been several coding techniques proposed in literature for this purpose as in [8], [9], [10], [11].

Cyclic codes have an advantage over other types of codes that they are easy to encode. Moreover, their well-defined mathematical structure led to very efficient decoding schemes [12]. For an (n, k) cyclic code it is uniquely verified with its generator polynomial $G(X)$ of order (n-k) from Galois field [12]. For our case study of (7, 4) cyclic code defined in [9], it has an ideal code generator polynomial given by

$$G(X) = 1 + X + X^3 \tag{7}$$

The parity check polynomial for this specific code as defined in [12] is given by:

$$h(X) = 1 + X + X^2 + X^4 \tag{8}$$

Then the parity check matrix is constructed from the parity check polynomial. The parity check matrix H is represented in its systematic form as [12]

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \tag{9}$$

Cyclic codes reduce the PAPR on the expense of increasing the bandwidth for the same data rate. But, this increase in bandwidth is small and comparable to single carrier methods [9]. Simulation results show that the BER performance of coded 4-PAM OFDM evaluated over AWGN outperforms the un-coded 4-PAM OFDM. Simulation results include the effect of white phase noise for different values of noise variances, for σ^2 = 0.02 rad^2 and for σ^2= 0.07 rad^2 as shown in Fig.6 respectively. Also, the effect of CFO on the same systems is investigated for normalized CFO= 0.05 and CFO= 0.1 as depicted in Fig. 7.

Hence the application of PAM-OFDM for high rate data links achieves the desirable effect of increasing the data rates with high spectral efficiency and bandwidth saving on the expense of increasing the PAPR. Further improvement in the achievable data rates, and in the reduction in the PAPR are achieved by the application of good coding techniques on the expense of smaller increase in bandwidth.

Nonlinearity effects introduced over AWGN channels, including phase noise and CFO degrade the BER performance, like other existing OFDM systems. There have been several methods proposed in literature like in [13] that compensate for the effect of phase noise and CFO and might be applied in our case for further improvement in the system performance.

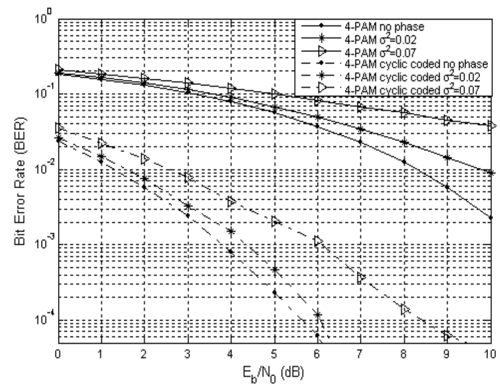


Fig. 6. BER comparison of coded and un-coded 4-PAM OFDM (128 carriers) over AWGN channel with white phase noise

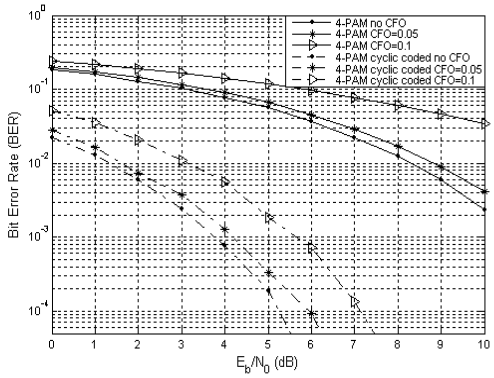


Fig.7. BER comparison of coded and un-coded 4-PAM OFDM (128 carriers) over AWGN channel impaired with CFO

VI. Conclusions

Recently, Orthogonal Frequency Division Multiplexing has been widely used in modern communication applications due to its spectral efficiency. In traditional wire-line communication PAM was used. The application of PAM for OFDM systems achieves bandwidth saving compared to M-ary QAM and M-ary PSK. This achievement is traded for power efficiency causing higher PAPR. In this paper the performance of M-ary PAM OFDM over AWGN channel with the impairment of phase noise and CFO is investigated with simulation. Furthermore, the proposal of coded 4PAM –OFDM for high rate data links is introduced.

Basically, the proposed system gains the high spectral efficiency associated with the application of OFDM with the achievement of the desired high rate. Furthermore, the application of a good coding technique achieves further improvement in the data rate and reduces the PAPR on the expense of smaller increase in the bandwidth for the same rate compared to the un-coded case. Finally, the performance of the proposed system over AWGN channel with phase noise and CFO was investigated in comparison with un-coded 4-PAM OFDM.

References

[1] Xiogen, F. "M-ary Amplitude Shift Keying OFDM System", *IEEE Trans. on Communications*, vol 51, NO 1,pp 1638-1642, Oct. 2003.

[2] L. Hanzo, C.H. Wong, M. S. Yee, *Adaptive Wireless Transceivers*. Wiley/IEEE press, 2002

[3] K. Sathananthan and C. Tellambura, "Probability of Error Calculation of OFDM Systems with Frequency Offset", *IEEE Trans. on Communications*, vol 49, NO 11,pp 1884-1888, Nov. 2001.

[4] Anna Garcia Armada, "Understanding the Effects of Phase Noise in Orthogonal Frequency Division Multiplexing (OFDM)", *IEEE Trans. on Broadcasting*, vol 47, NO 2,pp 153-159, June. 2001.

[5] W. H. Tranter, K. S. Shanmugan, T.S. Rappaport, K. L. Kusbar, *Principles of Communication Systems Simulation with Wireless Applications*. Prentice Hall, 2004.

[6] Bessios, A.G.; Stonecypher, W.F.; Agarwal, A.; Zerbe, J.L.," Transition-limiting codes for 4-PAM signaling in high speed serial links", *Global Telecommunications Conference, GLOBECOM '03*. IEEE Volume 7, 1-5 Dec. 2003 Page(s):3747 - 3751 vol.7, 2003.

[7] Farzan, K.; Johns, D.A., "A power-efficient 4-PAM signaling scheme with convolutional encoder in space for chip-to-chip communication", *ESSCIRC Proceeding of the 30th European Solid-State Circuits Conference*, Page(s):315 – 318, 21-23 Sept. 2004.

[8] Olfat, M.; Liu, K.J.R, "Low peak to average power ratio cyclic Golay sequences for OFDM systems", *IEEE International Conference on Communications*, Volume 2, Page(s):993 – 997 Vol.2, 20-24 June 2004.

[9] Do Horng Guo; Chau-Yun Hsu, "Systematic reducing the PAPR of OFDM by cyclic coding", *APCC 2003. The 9th Asia-Pacific Conference on Communications*, Volume 1, Page(s):133 - 137 Vol.1, 21-24 Sept. 2003.

[10] J. Jones, T.A. Wilkinson, and S.K. Barton, "Block coding scheme for reduction of peak to mean envelope power ratio of multicarrier transmission schemes", in *Electronics letters*, 30, 25, pp 2098-2099, 1994.

[11] S. Shepherd, P. van Eetvelt, C. Wyatt-Millington, and S.K. Barton, " Simple coding scheme to reduce peak factor in QPSK multicarrier modulation", *Electronics Letters*, vol. 31, pp. 1131-1132, July 1995.

[12] Haykin S., *Digital Communications*. John Wiley & Sons, 1988.

[13] L. Hanzo, A. Webb, and T. Keller, *Single- and Multi-Carrier Quadrature Amplitude Modulation*. New York: John Wiley-IEEE Press, April 2000.

Index