Produktion und Logistik

RESEARCH

Xin Wang

Operational Transportation Planning of Modern Freight Forwarding Companies

Vehicle Routing under Consideration of Subcontracting and Request Exchange

Springer Gabler

Produktion und Logistik

Herausgegeben von
B. Fleischmann, Augsburg, Deutschland
M. Grunow, München, Deutschland
H.-O. Günther, Berlin, Deutschland
S. Helber, Hannover, Deutschland
K. Inderfurth, Magdeburg, Deutschland
H. Kopfer, Bremen, Deutschland
H. Meyr, Hohenheim, Deutschland
Th. S. Spengler, Braunschweig, Deutschland
H. Stadtler, Hamburg, Deutschland
H. Tempelmeier, Köln, Deutschland
G. Wäscher, Magdeburg, Deutschland

Diese Reihe dient der Veröffentlichung neuer Forschungsergebnisse auf den Gebieten der Produktion und Logistik. Aufgenommen werden vor allem herausragende quantitativ orientierte Dissertationen und Habilitationsschriften. Die Publikationen vermitteln innovative Beiträge zur Lösung praktischer Anwendungsprobleme der Produktion und Logistik unter Einsatz quantitativer Methoden und moderner Informationstechnologie.

Kontakt

Professor Dr. Thomas S. Spengler
Technische Universität Braunschweig
Institut für Automobilwirtschaft
und Industrielle Produktion
Katharinenstraße 3
38106 Braunschweig

Xin Wang

Operational Transportation Planning of Modern Freight Forwarding Companies

Vehicle Routing under Consideration of Subcontracting and Request Exchange

Foreword by Prof. Dr.-Ing. Herbert Kopfer

Springer Gabler

Xin Wang
Bremen, Germany

Dissertation University of Bremen, 2014

ISBN 978-3-658-06868-4 ISBN 978-3-658-06869-1 (eBook)
DOI 10.1007/978-3-658-06869-1

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Library of Congress Control Number: 2014948094

Springer Gabler

Printed on acid-free paper

Springer Gabler is a brand of Springer DE.
Springer DE is part of Springer Science+Business Media.
www.springer-gabler.de

Foreword

Road haulage is the only mode of transportation that can be flexibly used to serve individual customers in wide and rural areas at reasonable cost. Especially in times of increasing virtualization of retail relationships through online-shopping, it is expected that the continuously rising transportation volumes will mainly be carried out by road haulage. However, the "limits to growth" of road transportation are already noticeable through phenomena like traffic congestion, increased pollution, etc. Moreover, the market suffers under persistently falling prices, which are mostly due to the liberalization of European markets.

Although freight forwarding companies in the road haulage are confronted with the pressure of both increasing costs and falling revenues, there are lots of unused capacities in the transportation market, which can directly be observed by considering the amount of partially loaded or even empty truck movements. Such unused capacities represent an important potential for increasing the efficiency of the entire transportation sector. A considerable increase of efficiency can be reached by vertical and horizontal cooperation in the transportation sector, which actually means to exploit and combine all three transportation modes: self-fulfillment, subcontracting and request exchange in coalitions of carriers.

Vertical collaboration is a well-known and important medium for short-term capacity expansion at variable costs in road haulage. Transportation requests can be subcontracted to other carriers who are paid according to specific tariffs. Despite the high practical importance of subcontracting, transportation planning considering this option of request fulfillment is still insufficiently investigated in literature.

Further potential for even higher efficiency can be realized through joint operational planning and request fulfillment within horizontal coalitions of freight forwarders. However, developing approaches for collaborative transportation planning of forwarders cooperating on the basis of equal partnership is a difficult task. The underlying planning tasks are extremely complex and, even more complicating, the general practical conditions like the tendency and willingness of forwarders to hide planning-relevant information against other partners in the coalition must be respected.

The research of this book focuses on the planning tasks of freight forwarding companies which are combining the vehicle routing and scheduling of own vehicles with the usage of external transportation capacities arising from vertical and horizontal cooperation with other freight carriers. The objective of the research is to develop mechanisms which enable the realization of almost all of the theoretically achievable potential of efficiency increase that can be reached by including collaboration in vehicle routing; i.e., by integrating the three possible fulfillment modes self-fulfillment, subcontracting, and collaborative request exchange to a simultaneous way of integrated operational collaborative transportation planning for road transport.

This book contains important findings of both practical and theoretical relevance for the research field of transportation planning. From a practical view, it provides efficient methods for the resolution of the highly important problems of the integrated transportation planning and the collaborative transportation planning, as well as the combination of them. From a scientific view, the extension of the vehicle routing problem, which has been intensively studied since many years and is still being intensively studied nowadays, through the consideration of practice-relevant aspects of subcontracting and collaboration broadens the spectrum of the research on transportation planning.

This book should be most suitable to researchers and students of logistics, particularly those with an engineering background. In addition, the contents of this book might be very interesting to those in industry who need to solve problems on the design, operation, and management of freight transportation.

Herbert Kopfer

Preface

This thesis represents the result of my PhD study at the Chair of Logistics, University of Bremen from September 2008 to January 2014. These years have been a long and challenging journey, but I was fortunate enough to be accompanied by many people. Without their support I would have never finished my dissertation. I thank all of them, particularly the ones below.

First, I would like to express my great gratitude to my supervisor Prof. Herbert Kopfer. He offered me the opportunity to work at the Chair of Logistics and to conduct my PhD research. He was very open-minded in his supervision and I enjoyed very much the freedom he gave me to pursue the research direction I like. Throughout my study, he continuously supported me with his broad knowledge in the fields of logistics and operations research, and he inspired me with his enthusiasm for new problems and solution ideas. His professional comments and invaluable advices have helped me improve the scientific quality of this thesis.

My great gratitude goes also to PD Dr. Jörn Schöberger who is the second reviewer of my PhD thesis. The considerable discussions we had and the constructive suggestions he made on my work were very helpful. I would also like to thank Prof. Herbert Kotzab and Prof. Jan Dethloff for being my examiners in my PhD defense and thank Prof. Tobias Buer and Mario Ziebuhr for being members of the doctoral committee.

I am thankful to my colleagues from the Chair of Logistics Julia Funke, Mario Ziebuhr and Kristian Schopka for helping me with the proofreading of my thesis and for participating in my defense. My appreciation also goes to my former colleagues, especially Dr. Christoph Manuel Meyer, Dr. Marta Krajewska, and Dr. Sebastian Sterzik, for their help and the nice time we had together.

I would like to thank Prof. Stefan Ropke for the discussions we had during the time he visited the Chair of Logistics in 2008, which inspired me a lot. Furthermore, I thank Prof. Michel Gendreau for his hospitality during the time Prof. Kopfer, Prof. Buer and I visited him in Montréal in October 2011 and for our fruitful collaboration.

I am grateful to the editorial team of Gabler Verlag for publishing my PhD thesis, especially Prof. Thomas Stefan Spengler, Prof. Hans-Otto Günther, Ms. Marta Schmidt, and Ms. Nicole Schweitzer.

My special thanks go to my family, who have unconditionally supported me all this time. I thank my mother Shuzhi Wang and my father Jiaji Wang for their love and encouragement. The thought that I could always rely on them gave me great confidence in overcoming any obstacles on my journey. Apart from that, my mother took the time and effort to proofread the entire manuscript of my PhD thesis. Furthermore, I would like to thank my mother-in-law Yun Shi and father-in-law Li Yin for their confidence in me. My wife Shanshan Yin deserves a special word of appreciation for her moral support, for her patience and love. It is a great fortune to have her on my side.

Last but not least, I would like to thank my daughter Xinghao Olivia for the excitement and happiness she brought me. I will never forget the moments when I was writing this thesis on the computer and she, who was only one year old at that time, came to me, climbed up my legs with assistance, and "helped" me by simply typing any randomly chosen characters. Such sweet actions of hers were the best I have got to relieve the pressures.

Xin Wang

Contents

List of Figures

List of Tables

Abbreviations

ALNS	Adaptive Large Neighborhood Search
B&B	Branch-and-Bound
CA	Combinatorial Auction
CP	Centralized Planning
CTP	Collaborative Transportation Planning
DARP	Dial-A-Ride Problem
DCFPDPF	Dynamic Collaborative Full-Truckload Pickup and Delivery Problem with Forwarding
DOD	Degree of Dynamism
DP	Dynamic Programming
DPDP	Dynamic Pickup and Delivery Problem
DU	Distance Unit
EDOD	Effective Degree of Dynamism
EEDOD	Extended Effective Degree of Dynamism
FCP	Freight Consolidation Problem
FLP	Forward-Looking Planning
FTL	Full-Truckload
HVRP	Heterogeneous Vehicle Routing Problem
IOTP	Integrated Operational Transportation Planning
IOTPP	Integrated Operational Transportation Planning Problem
IP	Isolated Planning
LNS	Large Neighborhood Search
LP	Linear Programming
LS	Local Search
LTL	Less-Than-Truckload
MIP	Mixed Integer Programming
MU	Monetary Unit
MYP	Myopic Planning
O-D	Origin-Destination
PDP	Pickup and Delivery Problem
PDPTW	Pickup and Delivery Problem with Time Windows

SA	Simulated Annealing
SCP	Set Covering Problem
SPP	Set Partitioning Problem
TS	Tabu Search
TSP	Traveling Salesman Problem
TSPP	Transportation Service Procurement Problem
TSPTW	Traveling Salesman Problem with Time Windows
VRFP	Vehicle Routing and Forwarding Problem
VRP	Vehicle Routing Problem
VRPB	Vehicle Routing Problem with Backhauls
VRPPD	Vehicle Routing Problem with Pickup and Delivery
VRPTW	Vehicle Routing Problem with Time Windows
WDP	Winner Determination Problem

1 Introduction

The increasing pressure on modern freight forwarding companies to improve profitability as a result of ever-changing conditions and fierce competition on the transportation market has strongly influenced their fleet management and transportation planning strategies. In order to utilize transportation resources more efficiently, it is no longer sufficient for freight forwarders to optimize the usage of their internal resources, but they also have to improve the management of external relations with other carriers to exploit more cost-saving potential embedded in cooperation. A holistic consideration of all possible options of request fulfillment makes the operational transportation planning of modern freight forwarders extremely complicated and new planning techniques have to be developed to help them realize this cost-saving potential.

1.1 Request fulfillment of freight forwarding companies

A transportation request can be generally specified by a pickup location, a delivery location, and a load of goods to be transported. There may be more restrictions related to a request. For instance, a time window for a certain operation (pickup or delivery) defines the time slot, in which the operation must be started. The task of freight forwarders is to fulfill their customer requests in a cost efficient way and to make sure that the customer requirements on service quality are met.

For the fulfillment of their acquired customer requests, forwarders can apply the following options:

1. keeping the execution in-house using their own fleet,
2. forwarding requests to subcontractors of different types, and
3. exchanging requests with partners in horizontal coalitions.

The first option is referred to as *self-fulfillment*. Using exclusively the own vehicles for the request fulfillment has been the main topic of research on vehicle routing problems, which have been extensively studied over the past decades. The task of the planning is to construct a set of routes and assign them to available vehicles so that all customers are served. A route is a sequence of customer locations which starts and ends at specific depots. The primary objective is to minimize the number of used vehicles and the secondary objective is to minimize the total travel distance or total costs of the routes.

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_1,

The second option is called as *subcontracting* and is widely used by forwarders in practice to increase the flexibility of their transportation capacity to deal with the fluctuating demand on the transportation market. Due to the high fixed costs of vehicles, many forwarders strongly downsize their own fleet to reach a maximal utilization of the own vehicles. Usually, only a part of the acquired requests can be fulfilled using the own transportation resources while the remaining requests are outsourced. Forwarders have to pay for the services brought by their subcontractors for the fulfillment of the outsourced requests.

In addition to subcontracting, freight forwarding companies can also build up horizontal coalitions with other fellow forwarders and perform *request exchange*. The motivation of considering this option lies in the cost-saving potential embedded in horizontal cooperation. Through request exchange, forwarders can consolidate complimentary requests of different coalition members and construct more efficient vehicle routes to reduce the total costs of the entire coalition. The obtained cost-savings present the joint benefits of the horizontal coalition that cannot be achieved by forwarders individually. The profitability of participating forwarders can then be further improved by acquiring their shares of the joint benefits.

Although both *subcontracting* and *request exchange* deal with transferring requests to other cooperation forwarders, these two options are essentially different. Subcontracting happens in vertical cooperation, while the forwarders and their subcontractors are in a hierarchical relation and forwarders plan independently for their internal and external capacities without explicitly coordinating with their subcontractors. On the contrary, request exchange is performed on the basis of an equal partnership of the members in horizontal coalitions. Besides planning for themselves (i.e. for their own fleet and, if applicable, for their subcontractors either), all partners must try to harmonize their plans with those of other coalition members.

Compared with the vehicle routing and scheduling problems, the research on transportation planning using different options of request fulfillment is much less despite of its high practical relevance. Especially, the topic of a simultaneous consideration of all three options in the operational planning is still rarely studied. Moreover, an integrated planning can be used in both static and dynamic situations. In a static planning, all information needed for the planning is available to the planner. In a dynamic environment, however, customer requests are released during the entire time horizon and the planner has to deal with the constant actualization of the status of the request portfolio and the vehicles. This fact makes the consideration of the dynamic planning process a quite challenging task in the academic research on transportation logistics. This thesis is thus dedicated to gain an insight into the operational transportation planning of modern freight forwarding companies in both static and dynamic environments and to appeal to more intensive studies in this research area.

1.2 Objectives and Structure of the Thesis

The options of fulfilling customer requests in the context of vertical and horizontal cooperation offer freight forwarding companies new possibilities to reduce their operational costs and thus improve their profitability on the much more competitive transportation market nowadays. However, due to the high heterogeneity of these different options, it is a great challenge to offer new planning techniques that can help forwarders realize the cost-saving potential embedded in cooperation. The objective of this thesis is thus to design solution approaches for the optimization problems that arise in the integrated consideration of applying more request fulfillment options simultaneously in the operation planning of freight forwarders. These approaches should help them reduce their total fulfillment costs and increase the operational efficiency of transportation resources as much as possible.

The thesis is organized as follows. In Chapters 2 and 3, the basic optimization problems to be solved in the operational transportation planning are introduced. The first type of these problems is the vehicle routing, which is the topic to be discussed in Chapter 2. These problems need to be solved if the forwarders have to directly bear all expense of the vehicle routes they have planned. Since this expense depends strongly on the total length of the routes and the total working time of the drivers, the general objective of these problems is to construct the most efficient routes while ensuring that all customers are served. The discussion focuses on the pickup and delivery problem (PDP) in a general form, for this problem is considered as the basic routing problem in the operational transportation planning that will be studied in later chapters.

Chapter 3 deals with the second type of the basic decision problems in the operational transportation planning which is called as freight consolidation. This problem needs to be solved when forwarders are outsourcing requests to other carriers but the forwarders themselves are not in charge of making concrete fulfillment plans for their subcontractors. In this case, the forwarders pay for the transportation services they “buy” from their subcontractors, while the subcontractors are responsible for generating concrete routing plans for the fulfillment of both the outsourced requests and the subcontractors’ own requests.

In Chapter 4, the extension of vehicle routing by subcontracting is discussed. Considering self-fulfillment and subcontracting, and disposing both own and subcontractors’ vehicles enable freight forwarders to reduce their fleet size. As a consequence, the process of forwarding requests to subcontractors has to be integrated into the traditional vehicle routing and scheduling. A rigorous reduction of the own fleet enables forwarders to do "cherry-picking", i.e., assigning the most profitable tours to their own vehicles and filling the gap between the own fleet capacity and customer demands by using capacities of subcontractors. But simply doing "cherry-picking" does not really make sense unless the outsourced requests are executed in a cost-efficient way as well. In order to minimize the overall total fulfillment costs including both the costs of the routes for the own vehicles and the costs for subcontracting, both options have to be considered in the operational transportation planning in a simultaneous way, which is referred to as the integrated operational transportation planning (IOTP) according to Krajewska (2008). The resulted optimization problem which combines the problems of *mode assignment* (self-fulfillment or subcontracting) and *vehicle routing* is called as the IOTP problem (IOTPP) and presented at the end of this chapter.

In order to solve the IOTPP presented in Chapter 4, two novel solution approaches are proposed in Chapter 5. The first one is an adaptive large neighborhood search (ALNS) heuristic and is developed based on the ALNS heuristic of Ropke and Pisinger (2006), which belongs to the most powerful heuristics designed for solving the PDP (Parragh et al., 2008b). The second approach uses the new ALNS heuristic iteratively to find efficient vehicle routes and recombines them to generate high quality IOTPP solutions.

Chapter 6 deals with the combination of self-fulfillment and request exchange and addresses the planning problems of freight forwarding companies within horizontal coalitions. In order to realize the cost-saving potential embedded in request exchange, cooperating forwarders can apply centralized planning by transferring all request information and decision-making competences to an authority of the coalition, who construct a single plan for all coalition members. In case that coalition members want to preserve their private information and autonomy of decision-making, decentralized planning approaches with complicated mechanisms for the harmonization of individual plans have to be developed. This decentralized planning is referred to as collaborative transportation planning (CTP) (Wang and Kopfer, 2014) and modeled in this chapter.

A *route-based request exchange mechanism* is then presented in Chapter 7 for the CTP of forwarders exchanging less-than-truckload (LTL) pickup and delivery requests with time windows. This new approach is developed following the decomposition principle of Dantzig and Wolfe (1960) under the consideration of the realistic restriction that all collaborating partners have only limited capacities in their own fleets. Computational study shows that this approach is very efficient and effective in terms of realizing potential cost-savings embedded in CTP while the exposure of private information is limited and the autonomy of the coalition members is preserved.

The studies on the operational transportation planning of forwarders in both vertical and horizontal cooperation, i.e., the IOTP and CTP, make it possible to study a more general case in Chapter 8, where all available resources, i.e., the own vehicle fleet, capacities of subcontractors including the common carriers, and the capacities of collaboration partners in horizontal coalitions through request exchange, are considered systematically in the operational transportation planning. The approach presented in Chapter 7 is then extended to solve the complex problem and validated by a computational study.

In Chapter 9, the operational transportation planning of freight forwarders is studied in a dynamic environment. Two approaches are developed to solve the dynamic problem by integrating the extended route-based request exchange mechanism presented in Chapter 8 into two rolling horizon planning frameworks.

In Chapter 10, a comprehensive computational study including five tests is conducted to have a more clear and deeper perception of the dynamic version of the CTP based on the discussion in Chapter 9. It is analyzed how the planning results can be improved through CTP and how the realized cost-savings are influenced by diverse factors of the situation and by adjusting the planning settings.

Chapter 11 summarizes the main findings of this thesis and outlines future research perspectives for the study on operational transportation planning of freight forwarding companies.

2 Vehicle Routing

In Chapter 2 and Chapter 3, the basic decision problems in the operational transportation planning of freight forwarding companies are introduced. If own disposable vehicles are used for the request fulfillment, the forwarders have to solve the vehicle routing and scheduling problems. The requests have to be assigned to different trucks, and for each truck, the order of the visits at the customer nodes assigned to this truck has to be specified. This topic is discussed in this chapter. The focus of the discussion lies in the PDP, which is considered as the basic routing problem for the study of the transportation planning in the context of IOTP and CTP in later chapters.

The problems of vehicle routing belong to the most studied combinatorial optimization problems. A great number of specific problems in this problem family have been discussed to which numerous publications are dedicated. Besides transportation of commercial goods, these research efforts embrace a broad spectrum of real-world applications including, for instance, solid waste collection, street cleaning, school bus routing, dial-a-ride systems, and transportation of handicapped persons.

This chapter focuses on some basic static routing problems. We begin with a brief introduction on this topic with some illustrative examples. Then, a mathematical formulation is presented to model the PDP with time windows (PDTPW). Next, a short overview of the algorithms proposed for the PDPTW is given. Finally, the ALNS heuristic for the PDPTW proposed by Ropke and Pisinger (2006) is described. This heuristic will be used and further developed in the forthcoming chapters to solve the PDPTW and some related extended routing problems in the context of IOTP and CTP.

2.1 Introduction

The simplest problem of vehicle routing is the traveling salesman problem (TSP), which deals with the following question: Given a set of cities a salesman has to visit and the distances between each pair of these cities, what is the shortest route that visits each city exactly once and returns to the origin city? This problem is one of the most famous NP-hard problems in combinatorial optimization. According to Müller-Merbach (1983), the research on the TSP can be traced back to 1832, when the problem was formulated for the first time in a German manual, in which five tours through Germany are suggested and

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_2,

one of them also through some Swiss cities. Figure 2.1 shows one of these five tours along 45 German cities, which is 1285 km long[1]. Schrijver (2005) studies briefly the history of the research on TSP till 1960. Detailed discussions about later research can be found in Lawler et al. (1985), Laporte (1992), Gutin and Punnen (2002), and Applegate et al. (2006).

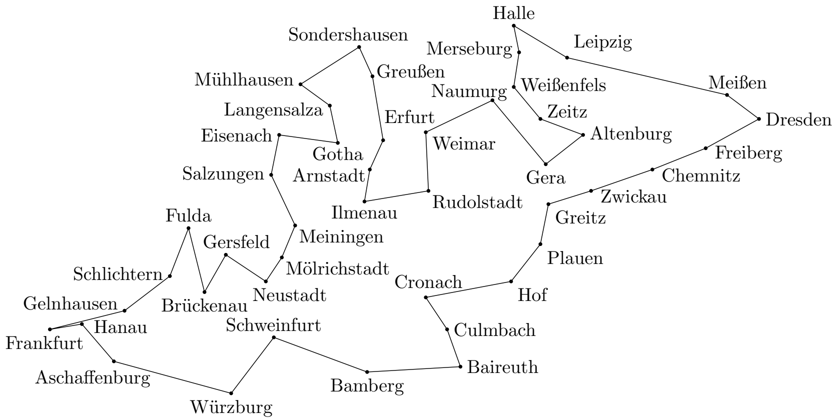


Figure 2.1: *A tour along 45 German cities suggested in 1832, cf. Schrijver (2005)*

Another very important and intensively studied problem is the vehicle routing problem (VRP) which was introduced by Dantzig and Ramser in 1959 (Dantzig and Ramser, 1959). Given a set of geographically scattered customers, to whom goods are to be delivered from the depot of a vehicle fleet, the VRP can be defined as the problem of constructing vehicle routes to serve all customers while the total distances of all routes are minimized. It must be assured that each customer is served exactly once by one vehicle that starts and ends its route at the depot and the total demands of customers served in a route do not exceed the capacity of the vehicle. The VRP can also be used to model the case when goods are to be gathered and transported to the depot, and the vehicles are empty when they start their routes. Figure 2.2 shows an example of the VRP with 4 vehicles and 14 customer nodes. The capacity of the vehicles is limited to 20 and the numbers around the customer nodes represent the demand of the customers.

Two closely related subproblems have to be considered by solving the VRP. The first one is to assign each customer to a specific vehicle and the second one is to construct a route for each vehicle that serves all customers assigned to it. Thus, the VRP presents

[1]The shortest tour with a total length of 1248 km can be found for this instance. However, if the local conditions are taken into account, this 1832 tour might be optimum. (Schrijver, 2005)

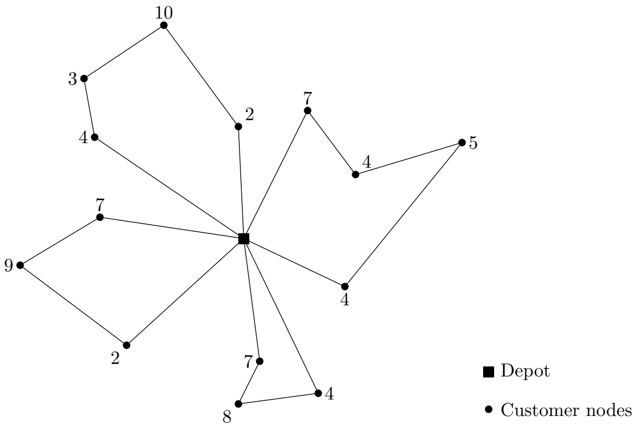
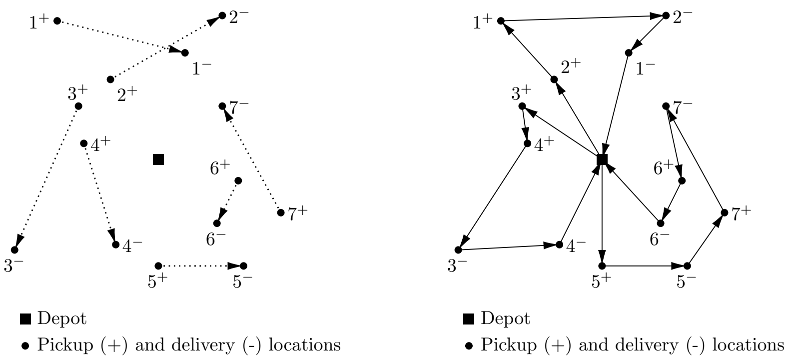


Figure 2.2: *An example of the VRP*

a generalization of the TSP and is therefore an NP-hard problem. The TSP can be seen as a special case of the VRP while only one vehicle exists in the fleet and its capacity is large enough to serve the total demands of all customers. Since its introduction, a great number of scientific articles have been published. Recent reviews of the literature on the VRP can be found in Cordeau et al. (2007), Parragh et al. (2008a), and Laporte (2009). The two books edited by Toth and Vigo (2002a) and by Golden et al. (2008) offer a comprehensive introduction on the VRP and its most common variants as well as an extensive discussion about the solution methods.

In the classical VRP, all customer requests are of the same type, i.e., either delivery requests or pickup requests. A nature extension of this problem is the VRP with Backhauls (VRPB) which considers both request types. In the VRPB, the customer set is partitioned into two subsets. The first subset contains the *linehaul customers* (delivery requests) and each of them requires a given load to be transported from the depot to the customer node. The second one contains the *backhaul requests* (pickup requests), where a given quantity of goods have to be picked up at the customer node and transported back to the depot. A practical restriction considered additionally is that on each route, the backhaul customers, if any, are visited after all linehaul customers, for the fact that vehicles are often rear-loaded and the on-board load rearrangement required by a mixed service is difficult or even impossible at customer locations (Toth and Vigo, 2002b).

In the VRP and the VRPB, the transportation of goods between customer locations and a depot is studied. A further extension is the PDP, which considers the transportation of goods or persons between pickup and delivery locations without transshipment at intermediate locations[2]. Since both the pickup and delivery locations related to a request need not be the depot, the PDP is a generalization of the VRP and the VRPB. If the pickup location (or the delivery location) of all requests is the depot, the PDP returns to the VRP. If either the pickup or the delivery location of each single request is the depot, the PDP goes back to the VRPB. Some recent surveys on the PDP can be found in Desaulniers et al. (2002), Berbeglia et al. (2007), Parragh et al. (2008b), and Cordeau et al. (2008). Figure 2.3 shows an example of the PDP with 7 homogeneous LTL requests (presented by the arrow with dotted lines in Figure 2.3a) and one of its possible solutions with 3 vehicles (depicted by the arrows with solid lines in Figure 2.3b).



(a) Requests **(b)** A solution

Figure 2.3: *An example of the PDP*

The routing problems can be further extended by introducing the time window at the locations, which is defined as a time interval, in which the needed operation at the location must be started. Time windows can be either soft or hard. Soft time windows can be violated at a penalty cost, while hard time windows must be strictly held. In case of hard windows, vehicles can only start the services within the associated time windows. In other words, they have to wait at the customer locations if they arrive before the customers are ready for the service. The extensions of the TSP, the VRP, and the PDP by time windows can be referred to as the TSP with time windows (TSPTW), the VRP with time windows (VRPTW), and the PDPTW, respectively. These problems can be further extended by multiple depots. In the most general case of the PDP, each vehicle can have its own start and end depots. In the following section, a mathematical model is presented for this case.

[2]The PDP is also referred to as the VRP with pickup and delivery (VRPPD) in, for instance, Desaulniers et al. (2002) and Parragh et al. (2008b). In this thesis, the abbreviation "PDP" is used to denote this problem.

2.2 Mathematical model for the PDPTW

In the PDPTW, a set of transportation requests have to be fulfilled using a vehicle fleet. Each request is specified by a pickup location, a delivery location, and a load to be transported between these two locations. Moreover, a customer payment will be paid for the fulfillment of a request. The objects of transportation related to the requests can be either goods or persons. Specifically, the problem of transporting persons is referred to as the dial-a-ride problem (DARP) in which the convenience of the people is usually considered in the objective function. The service at each location must be started within a time window defined by the customer and is associated with a service time. If a vehicle arrives at a location prior to the beginning of the time window, it has to wait. The service time indicates how long it will take to finish the service. Furthermore, some requests may require special equipment and thus can be only served by specific vehicles in the fleet. The vehicles for the request fulfillment can be heterogeneous. They can have different start and end depots, different capacities, and can be differently equipped. Also the costs related to the vehicles can vary from each other.

The task of the PDPTW is to construct a set of valid vehicle routes to serve all the requests. A route for a vehicle is a sequence of locations, which begins at the vehicle's start depot and terminates at its end depot. It is valid when:

1. each customer location in this route is visited exactly once,
2. both the pickup and the corresponding delivery of a request assigned to this vehicle are served in this route,
3. the pickup location of a request must be visited before its corresponding delivery location,
4. the capacity of the vehicle is not exceeded, and
5. the compatibility constraints are satisfied.

The objective of the PDPTW is to minimize the total fulfillment costs. As all customers must be served and the customer payments are fixed, minimizing the total costs is equivalent to maximizing the total revenue, determined as the difference between the total customer payments and the total fulfillment costs.

The PDPTW can be mathematically formulated based on Desaulniers et al. (2002) and Ropke and Pisinger (2006) in the following way. Let $R = \{1, 2, \ldots, n\}$ be the set of all requests. Define the set of pickup nodes as $P = R = \{1, 2, \ldots, n\}$ and the set of delivery nodes as $D = \{n + 1, n + 2, \ldots, 2n\}$. Since $P = R$, the origin-destination (O-D) pair of a request $r \in R$, (r^+, r^-), can be specified as $(u, u + n)$, $u = r \in P$. It is possible that different nodes represent the same geographical location. Denote the transportation load required by request r as ℓ_r. The capacity requirement at the pickup node of a request $r \in R$ can be defined as $\ell_u = \ell_r$, $u = r$, and at its corresponding delivery node as $\ell_{u+n} = -\ell_r$, $u = r$. The time window at node $u \in P \cup D$ is given by $[b_u, e_u]$ and the service time is defined as s_u.

Let K be the vehicle set including ϱ vehicles. As not all nodes can be served by all vehicles in K, two specific sets of pickup nodes P_k and delivery nodes D_k corresponding to the requests that can be served by a vehicle $k \in K$ can be defined, while $P_k \subseteq P$ and $D_k \subseteq D$. The start and end depots of a vehicle k are denoted as o_k and o'_k, respectively. A graph of the entire problem can be defined as $G = (V, A)$, where $V = P \cup D \cup \{o_1, o_2, \cdots, o_\varrho\} \cup \{o'_1, o'_2, \cdots, o'_\varrho\}$ is the node set and $A = V \times V$ is the arc set. The distance and the travel time of an arc $(u, v) \in A$ are given by d_{uv} and d'_{uv}, respectively. For each vehicle k, a graph $G_k = (V_k, A_k)$ can be defined in the same way as G with $V_k = P_k \cup D_k \cup \{o_k, o'_k\}$ and $A_k = V_k \times V_k$. Each vehicle $k \in K$ has a limited capacity Q_k and is associated with a fixed cost α_k, a cost rate per distance unit (DU) β_k, and a cost rate per time unit β'_k. It is further assumed that vehicle $k \in K$ leaves its start depot o_k without any load when its time window opens at b_{o_k} and $e_{o_k} = b_{o_k}$. The service time at the depots s_{o_k} and $s_{o'_k}$, $k \in K$ is defined as 0.

Three decision variables are needed in the formulation. The binary variable x_{uvk}, $u, v \in V_k$, $k \in K$, equals to 1 if vehicle k travels from node u to node v, and 0 otherwise. Variable t_{uk} defines the time when the service at node $u \in V_k$ starts by using vehicle $k \in K$. Variable l_{uk} gives the load of vehicle $k \in K$ after the service at node $u \in V_k$ is completed. Both variables t_{uk} and l_{uk} are only well-defined when node u is actually served by vehicle k. The PDPTW can be modeled as follows:

$$\min \sum_{k \in K} \alpha_k + \sum_{k \in K} \sum_{(u,v) \in A_k} \beta_k d_{uv} x_{uvk} + \sum_{k \in K} \beta'_k (t_{o'_k k} - b_{o_k}) \tag{2.1}$$

subject to:

$$\sum_{k \in K} \sum_{v \in V_k \setminus \{o_k\}} x_{uvk} = 1 \quad \forall u \in P \tag{2.2}$$

$$\sum_{v \in P_k \cup D_k} x_{uvk} - \sum_{v \in P_k \cup D_k} x_{v,n+u,k} = 0 \quad \forall k \in K, u \in P_k \tag{2.3}$$

$$\sum_{v \in P_k \cup \{o'_k\}} x_{o_k,v,k} = 1 \quad \forall k \in K \tag{2.4}$$

$$\sum_{u \in D_k \cup \{o_k\}} x_{u,o'_k,k} = 1 \quad \forall k \in K \tag{2.5}$$

$$\sum_{u \in V_k \setminus \{o'_k\}} x_{uvk} - \sum_{u \in V_k \setminus \{o_k\}} x_{vuk} = 0 \quad \forall k \in K, v \in P_k \cup D_k \tag{2.6}$$

$$x_{uvk}(t_{uk} + s_u + d'_{uv} - t_{vk}) \leq 0 \quad \forall k \in K, (u,v) \in A_k \tag{2.7}$$

$$b_u \leq t_{uk} \leq e_u \quad \forall k \in K, u \in V_k \tag{2.8}$$

$$t_{uk} + d'_{u,n+u,k} \leq t_{n+u,k} \quad \forall k \in K, u \in P_k \tag{2.9}$$

$$x_{uvk}(l_{uk} + \ell_v - l_{vk}) = 0 \quad \forall k \in K, (u,v) \in A_k \tag{2.10}$$

$$\ell_u \leq l_{uk} \leq Q_k \quad \forall k \in K, u \in P_k \tag{2.11}$$

$$0 \leq l_{n+u,k} \leq Q_k - \ell_u \quad \forall k \in K, u \in P_k \tag{2.12}$$

$$l_{o_k k} = 0 \quad k \in K \tag{2.13}$$

$$x_{uvk} \in \{0,1\} \quad \forall k \in K, (u,v) \in A_k \tag{2.14}$$

$$t_{uk} \geq 0 \quad \forall k \in K, u \in V_k \tag{2.15}$$

$$l_{uk} \geq 0 \quad \forall k \in K, u \in V_k \tag{2.16}$$

The objective function (2.1) minimizes the total fulfillment costs including both the fixed and variable costs, while the time dependent variable cost is calculated based on the total operating time of the vehicles. Because the fixed costs of vehicles are constant, the corresponding term in (2.1) can be omitted and an equivalent formulation of this objective function is given by:

$$\min \sum_{k \in K} \sum_{(u,v) \in A_k} \beta_k d_{uv} x_{uvk} + \sum_{k \in K} \beta'_k (t_{o'_k k} - b_{o_k}) \tag{2.17}$$

Constraint (2.2) imposes that each request is served by exactly one vehicle. Constraint (2.3) ensures that the corresponding pickup and delivery nodes are served by the same vehicle. Constraints (2.4) and (2.5) guarantee that each vehicle begins its route at its start depot and terminates it at its end depot. Constraint (2.6) is the flow balancing constraint and makes sure that if a vehicle serves a customer node, it has to leave it either. Constraints (2.7) and (2.8) determine the start time of the service at customer nodes, which must lie in the time window, while constraint (2.7) also eliminates any subtours. (2.9) ensures that the pickup is performed before its corresponding delivery. Constraints (2.10)-(2.13) make sure that the load variable is set correctly along the routes and the capacity of the vehicles is not exceeded. Moreover, constraints (2.7) and (2.10) in this model can be linearized by introducing a *big M* that is a large number:

$$t_{uk} + s_u + d'_{uv} - t_{vk} \leq (1 - x_{uvk})M \quad \forall k \in K, (u,v) \in A_k \tag{2.18}$$

$$l_{uk} + \ell_v - l_{vk} \leq (1 - x_{uvk})M \quad \forall k \in K, (u,v) \in A_k \tag{2.19}$$

For the homogeneous PDPTW, to minimize the number of total used vehicles is often considered as the primary objective, while to minimize the total variable costs of the routes as the objective function (2.17) does or just to minimize the total distances as the secondary one does. This means that a solution is better than another one when in the first solution, i.e., the better one, less vehicles are used to serve all requests or both solutions use the same number of vehicles but the total costs of the first one, especially the variable costs, are less than that of the second one. The usage of the lexicographic objectives is motivated by the fact that the fixed costs of a vehicle are usually much higher than the variable costs. Actually, these two objectives can be united into one single objective function by adding a fixed term into the objective function (2.17) by introducing a binary variable z_k, $k \in K$, which will be one if vehicle k is used or zero if not, and by setting the fixed costs related to the vehicles α_k very high. The united objective function for the homogeneous PDPTW can be formulated as:

$$\min \sum_{k \in K} \alpha_k z_k + \sum_{k \in K} \sum_{(u,v) \in A_k} \beta_k d_{uv} x_{uvk} + \sum_{k \in K} \beta'_k (t_{o'_k k} - b_{o_k}) \tag{2.20}$$

The following constraint can help decide whether a vehicle is used:

$$\sum_{v \in P_k} x_{o_k,v,k} \leq M \cdot z_k \quad \forall k \in K \tag{2.21}$$

In case that all vehicles are homogeneous, we have $\alpha_k = \alpha$, $P_k = P$, $o_k = o$, $o'_k = o'$, for all $k \in K$ and the index k can be removed from these notations.

2.3 Solution approaches for the PDPTW

The PDPTW is an NP-hard problem since it is a generalization of the VRP. Great efforts have been done in developing efficient algorithms to solve this problem and its variants in the last three decades. In this section, the algorithms proposed for the multi-vehicle PDP are briefly reviewed. The review is limited to the important contributions since 1990. An overview of the early stage of the research on the PDP from the early 1980s to the mid-1990s can be found in Savelsbergh and Sol (1995). For comprehensive reviews on the solution methods for a wider spectrum of PDP variants, the reader is referred to Berbeglia et al. (2007); Parragh et al. (2008b); Cordeau et al. (2008).

2.3.1 Exact algorithms

The very first exact algorithm proposed for the multi-vehicle PDP may be the algorithm of Dumas et al. (1991). The authors use the column generation scheme by applying a set partitioning formulation of the PDPTW. The subproblem of generating valid routes is modeled as a constrained shortest path problem and solved by a forward dynamic programming (DP) approach. Some small instances with up to 4 vehicles and 14 customers and 3 vehicles and 15 customers with tight time windows are solved to optimality. Sigurd et al. (2004) also use the column generation technique and develop an exact algorithm to solve a practical variant of the PDP considering additionally a precedence restriction associated with the requests. This additional restriction enables the authors to strongly reduce the computation complexity of the shortest path problem and to solve instances up to 580 nodes.

Another methodology that has proved successful for solving the PDP to optimality is the branch-and-cut which does not necessarily require tight constraints that can lead to a significant reduction of the solution space (Lu and Dessouky, 2004). Different branch-and-cut algorithms are proposed in Lu and Dessouky (2004) and Ropke et al. (2007) for the PDP and in Cordeau (2006) for the DARP. The algorithm of Lu and Dessouky (2004) is developed based on a formulation of the PDP with two-index flow variables. The authors also introduce a precedence variable in their formulation and identify several valid inequalities based on it. Instances with up to 5 vehicles and 25 customers were solved to optimality. Cordeau (2006) uses a three-index formulation of the DARP. Besides some inequalities adapted from existing algorithms for the TSP and the VRP, the author also proposes some new inequalities taking the specific structure of the problem into account. The algorithm was able to solve instances with up to 4 vehicles and 32 requests. In Ropke et al. (2007), two-index formulations of the PDPTW are used. However, the new formulations contain exponential number of constraints but fewer variables. The authors report better bounds than those obtained by previous algorithms. Instances with up to 8 vehicles and 96 requests were solved to optimality.

Motivated by the fact that set partitioning formulations of the VRP tend to provide stronger lower bounds than formulations based on flow variables (Bramel and Simchi-Levi, 2002) and by the success of the combined approaches of branch-and-price, i.e., the column generation scheme, and branch-and-cut in solving the VRP, Ropke and Cordeau (2009) propose a branch-and-cut-and-price algorithm for the PDPTW. Through perturbing the cost matrix of the pricing problem, fast algorithm can be used to solve the pricing problem with valid inequalities. Some large instances with up to 500 requests with tight time windows were solved using this algorithm. The latest exact algorithm for the PDPTW is proposed by Baldacci et al. (2011) based on a set-partitioning-like integer formulation. A bounding procedure is used to find a near-optimal dual solution to the linear programming (LP) relaxation of the formulation. The final dual solution is used to generate a reduced problem containing only selected routes, which is solved either by an integer programming solver or a branch-and-cut-and-price algorithm, depending on the size of this resulted problem. Computational results indicate a great improvement of the performance of this algorithm compared with the one of Ropke and Cordeau (2009).

2.3.2 Heuristic algorithms

Although the research on exact solution methods for the PDPTW has been put forward greatly in the last several years, some limitations still have to be recognized. For instance, only two of the algorithms described in Section 2.3.1, i.e., the algorithms of Lu and Dessouky (2004) and Sigurd et al. (2004) deal explicitly with the heterogeneity of vehicle fleet in terms of capacity and depots. In addition, no exact algorithm has been proposed to solve PDPTW instances when vehicles have different cost structures. Thus, efficient heuristics are still indispensable for solving more complex instances of the PDPTW.

Nanry and Barnes (2000) propose a reactive tabu search (TS) heuristic to solve the PDPTW. This heuristic begins with a greedy procedure for obtaining a feasible initial solution. Three distinct move neighborhoods that capitalize on the dominance of the precedence and coupling constraints are used in the following search process, which is directed by a hierarchical multi-neighborhood search strategy based on the average time window length of the instance.

A tabu-embedded simulated annealing (SA) heuristic (Dowsland, 1995) is developed by Li and Lim (2001) for the PDPTW. Each time a neighbor solution of the current one is accepted, it will be further improved through a descent local search (LS) algorithm. After some consecutive iterations without any improvement of the current solution, the heuristic restarts from the current best solution. A further contribution of this work is the generation of benchmark instance sets based on the VRP instance of Solomon (1987). The instance data and the best-known solutions can be found on the special website operated by SINTEF[3].

Similar neighborhood structures as defined in Nanry and Barnes (2000) are used in the TS heuristic proposed by Lau and Liang (2002). The authors use a partitioned insertion heuristic as the construction algorithm to generate initial feasible solutions, which combines the ideas of a simple insertion heuristic and an adapted version of the swap heuristic widely used for the VRP.

[3] `http://www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark`

Xu et al. (2003) propose a column generation based algorithm for a practical PDPTW, in which a series of restrictions commonly encountered in real-world applications including multiple time windows, working hours of drivers, and compatibility of requests are considered. A DP is used to solve the subproblem. A final solution is obtained by solving a restricted version of the set partitioning type formulation containing only the routes constructed by an integer programming solver during the column generation phase.

A grouping genetic algorithm is proposed by Pankratz (2005). This heuristic uses a group-oriented genetic encoding scheme and each gene represents a group of requests assigned to a single vehicle. Because the routing information within a gene is not directly handled by the genetic operators, additional data structures are used and a separate heuristic is called while decoding a chromosome.

In order to improve the quality of the initial solutions, an insertion-based construction heuristic is proposed by Lu and Dessouky (2006). Besides the distance increase, two new criteria are considered in the evaluation. The first one is the time window slack reduction which concerns the possibility of inserting more requests in the route in the following iterations. The second one is called as crossing length percentage, based on which a visual attractiveness is measured.

Bent and van Hentenryck (2006) propose a two-stage hybrid algorithm for the PDPTW. In the first stage, the number of total used vehicles is minimized using a simple SA heuristic. In the second stage a large neighborhood search (LNS) heuristic introduced in Shaw (1997) is used to minimize the total costs further. The neighborhood is defined as the set of solutions that can be reached by relocating up to a certain number of requests. The authors also report their results on some sets of the benchmark instances of Li and Lim (2001) in a previous article in Bent and van Hentenryck (2003).

Ropke and Pisinger (2006) extend the LNS heuristic by an adaptive mechanism and propose an ALNS heuristic for the PDPTW. This heuristic belongs to the most successful heuristic algorithms in solving the PDPTW benchmark instances of Li and Lim (2001) until present for the fact that it has found the best-known solutions for a great part of these instances, and primarily, the large instances. A more detailed description of this heuristic is given in the following section.

2.4 The ALNS heuristic of Ropke and Pisinger (2006)

In this section, the ALNS heuristic proposed by Ropke and Pisinger (2006) for the PDPTW is presented. Because of the success of this heuristic in the PDPTW literature, it is chosen as the solution methodology for the PDPTW and is further developed for the related problems in the context of IOTP and CTP considered in this thesis. It is not intended here to show every detail of this heuristic and the reader is thus referred to the original paper of Ropke and Pisinger (2006) for a more comprehensive description of the heuristic and its implementation, settings, and performances.

2.4.1 General ideas

That the ALNS heuristic contains several different types of removal and insertion operators enables a dynamic switch among these operators guided by the search process to achieve better results. Algorithm 1 gives an overview of the entire approach. In each iteration of the ALNS, one removal and one insertion operator is chosen (line 4) and used on the current solution s to generate a new solution s' in a large neighborhood (lines 5-7). The probability of applying a removal or an insertion operator is adapted during the search process (lines 14-15).

Algorithm 1 ALNS heuristic

```
1: initialize start solution s
2: s_best := s
3: while stopping criterion not met do
4:     select removal and insertion operators according to their weights
5:     s' := s
6:     remove requests from s' using the chosen removal operator
7:     reinsert requests into s' using the chosen insertion operator
8:     if f(s') < f(s_best) then
9:         s_best := s'
10:    end if
11:    if accept (s', s) then
12:        s =: s'
13:    end if
14:    if weight update criterion met then
15:        update weights of operators
16:    end if
17: end while
18: return s_best
```

Different from the earlier LNS heuristic in Shaw (1998) which uses a simple descent search strategy, the ALNS heuristic is embedded in an SA framework, which also accepts declined solutions in the neighborhood under circumstances to avoid being trapped at local optima. Algorithm 2 shows the general steps of an SA heuristic. Given an initial solution s, the local search process of the SA starts. In each iteration, a neighbor solution s' of the current solution s is generated. If s' is a better one than s (line 8), i.e., the objective function value $f(s')$ is smaller than $f(s)$ for a minimization problem, it will be accepted as the new current solution for the next iteration. If s' is worse than s, a probability value $e^{-\frac{f(s')-f(s)}{T}}$ is calculated based on the increment of the objective function value $f(s') - f(s)$ and the current *temperature* T in this iteration. A random value is generated in the range of $[0, 1]$ and compared with this probability. The neighbor solution s' will be accepted when the random value is smaller than the probability (line 10). The temperature decreases from the initial value T_0 every iteration according to a *cooling rate* ς, which lies in $(0, 1)$ (line 13). Thus, the possibility to accept the same decrement of the objective function value declines with the temperature and the algorithm gradually converges.

Algorithm 2 SA heuristic

1: initialize start solution s
2: $s_{best} := s$, $T := T_0$
3: **while** stopping criterion not met **do**
4:     generate a neighbor solution s'
5:     **if** $f(s') < f(s_{best})$ **then**
6:         $s_{best} := s'$
7:     **end if**
8:     **if** $f(s') \leq f(s)$ **then**
9:         $s := s'$
10:     **else if** $random[0, 1] < e^{-\frac{f(s')-f(s)}{T}}$ **then**
11:         $s := s'$
12:     **end if**
13:     $T := T \times \varsigma$
14: **end while**
15: return s_{best}

2.4.2 Removal operators

Three removal operators are used in the ALNS to remove a given number of requests from the current solution: *random removal*, *worst removal*, and the *Shaw removal.*

The first two removal operators are simple to follow. Random removal randomly chooses requests and removes them from vehicle routes. Worst removal iteratively removes the request with the highest marginal cost, which is calculated as the difference of the route costs with and without this request.

The Shaw removal is the most complicated one and originally proposed in Shaw (1997) and Shaw (1998). It systematically removes requests that are similar. The motivation of using this operator is that the more similar the removed requests are, the more options there are to switch these requests and thus the higher the possibility is to find better solutions. The similarity of two requests $r_1, r_2 \in R$ is evaluated according to a *relatedness measure* $R(r_1, r_2)$. The lower the measurement is, the more related, i.e., similar according to the current solution, these two requests are. Denote the pickup and delivery locations of two requests $r_1, r_2 \in R$ as r_1^+, r_2^+ and r_1^-, r_2^-, respectively. Define t_u as the time when service at the node $u \in P \cup D$ starts and K_r as the set of all vehicles that can serve request $r \in R$. The relatedness measure $R(r_1, r_2)$ can be calculated according to the following formula:

$$\begin{aligned} R(r_1, r_2) =& \phi(d_{r_1^+ r_2^+} + d_{r_1^- r_2^-}) + \chi(|t_{r_1^+} - t_{r_2^+}| + |t_{r_1^-} - t_{r_2^-}|) \\ &+ \psi|\ell_{r_1} - \ell_{r_2}| + \omega(1 - \frac{|K_{r_1} \cap K_{r_2}|}{\min\{|K_{r_1}|, |K_{r_2}|\}}) \end{aligned} \tag{2.22}$$

This measure consists of four term: a distance terms, a time term, a load term, and a vehicle compatibility term that considers the vehicles that can be used to serve the two requests. The last term can be omitted when all requests can be served by all vehicles. The authors also suggest normalizing d_{uv}, t_u, and ℓ_r, $u, v \in P \cup D, r \in R$ by scaling these parameters in the interval of $[0, 1]$, so that $0 \leq R(r_1, r_2) \leq 2(\phi + \chi) + \psi + \omega$. It must be mentioned that the relatedness measure of two specific requests may differ according to different solutions, because the routes in which these two requests are planned may be different and thus the time term may vary from each other.

By applying the worst removal and the Shaw removal, a randomness is additionally introduced. The purpose is to avoid situations where the same requests or the same pairs of requests are removed over and over again.

2.4.3 Insertion operators

Two parallel insertion operators are used in the ALNS heuristic. All these operators iteratively construct several vehicle routes at the same time. In each iteration, one request is inserted into the solution. Thus, they can directly be used as the construction heuristic for the ALNS to generate initial solutions.

The first one is called the *basic greedy heuristic*. This operator first evaluates the *insertion cost* c_{rk}, defined as the increment of the route cost after request r is inserted into route k at the best possible position in this route, for all requests that have not been yet planned in any route, and for all vehicle routes. If a request cannot be inserted into a route without violating any restriction, the insertion cost is set to $\Delta c_{rk} = \infty$. After that, the cost of inserting request r into the current solution can be defined as $\Delta c_r = \min_{k \in K} \Delta c_{rk}$. This is the cost of inserting this request into the solution at its best position overall. Then, the request with the smallest best insertion cost c_r among all requests that have not been inserted into any route is chosen and inserted into the solution at its best position overall. The process is repeated until no more insertions are possible.

The second type of insertion operators is the *regret heuristics*. Besides the evaluation of the direct insertion cost, they additionally estimate the consequence cost of not inserting a request in the current iteration. The request with the highest consequence cost is then inserted. After having evaluated the insertion cost Δc_{rk} for all requests not planned in any route, the regret heuristics calculate for each of these requests a *regret value* c_r^*, and insert the request with the largest regret value into the solution at its best position overall. Ties are broken by choosing the request with the lowest insertion cost c_r.

The regret value can be determined in the following way. Sort the insertion costs for each not planned request into different routes in the solution Δc_{rk} in ascending order. Then, we can use the variable $\zeta_{ri} = 1, 2, \ldots, |K|$ instead of the vehicle index k in Δc_{rk} to obtain the sorted list of the insertion cost as $< \Delta c_{r\zeta_{r1}}, \Delta c_{r\zeta_{r2}}, \ldots, \Delta c_{r\zeta_{ri}}, \ldots, \Delta c_{r\zeta_{r|K|}} >$. Thus, the i'th element in the list is the i'th smallest insertion cost of request r. The regret value c_r^* can be calculated in a *regret-q* heuristic as:

$$c_r^* = \sum_{i=1}^{q} (\Delta c_{r\zeta_{ri}} - \Delta c_{r\zeta_{r1}}) \tag{2.23}$$

If some requests cannot be inserted into at least $|K| - q + 1$ routes, then the request that can be inserted in the fewest number of routes but at least in one route is chosen and inserted at the best position overall. Ties are broken by choosing the request with the lowest best insertion cost c_r.

Due to the tie-breaking rules, the basic greedy insertion heuristic can be seen as the regret-1 heuristic. Specifically, so long as it is possible, the five heuristics: *regret-1*, *regret-2*, *regret-3*, *regret-4*, and *regret-*$|K|$ heuristics are used in the ALNS algorithm.

2.4.4 Adaptive operator choice

In each iteration of the ALNS heuristic, only one insertion and one removal operator are chosen and applied to the current solution. The *roulette wheel selection principle* is thus used for the operator choice. Suppose that there are m operators that can be chosen and the i'th operator is assigned a weight w_i, $i = 1, \ldots, m$. Then, an operator will be chosen with the probability $w_i / \sum_{i=1}^{m} w_i$. All operators are assigned the same weight at the very beginning of the search process.

Instead of using a predefined weight setting for all instances, an adaptive mechanism that dynamically adjusts the weights of the operators is used. To achieve this, the entire search process is divided into many *segments*, each including 100 iterations. Each operator has a score of 0 at the very beginning. This score will be increased by predefined amounts if (1) the last combination of removal and insertion operators has resulted in a new best solution, (2) the last combination of removal and insertion operators has resulted in a solution that has not been accepted before, and this new solution is better than the current solution, or (3) the last combination of removal and insertion operators has resulted in a solution that has not been accepted before, and this new solution is accepted despite that it is worse than the current solution. In each of the three cases both the removal and insertion operators are given the same score.

At the end of each segment, the weights are updated based on the recorded scores. Denote the weight of operator i in the s'th segment as w_{is}. The weight of operator i in the next segment $s+1$ is determined as:

$$w_{i,s+1} = w_{is}(1-\lambda) + \lambda \frac{\pi_i}{\theta_i} \tag{2.24}$$

where π_i is the score operator i won in the last segment, θ_i is the number of times it was attempted to use this operator in the last segment, and λ is a control parameter that determines how quickly the weight adjustment reacts to changes in the effectiveness of the operators.

2.4.5 Further settings

Since the ALNS heuristic uses an SA framework, the choice of a good cooling plan can have a great influence on its performance. Given the strategy of reducing temperature using the expression $T = T \cdot \varsigma$, it is important to choose a good start temperature T_0. The ALNS heuristic specifies the parameter for each single instance individually based on the initial solution generated by the construction heuristics which are the insertion operators. The start temperature is then set in such a way that the heuristic will accept with probability of 50% a solution that is 5% worse than the initial solution. The decline percentage parameter is calculated based on the total variable route costs. The bigger this parameter is, the slower the convergence of the SA process is. On the other hand, if this parameter is set too small, the ALNS heuristic can be trapped in local optima easily.

To introduce additional randomness in the search process and thus to improve the overall performance of the heuristic, a *noise term* is added in the objective function. Every time the insertion cost Δc_{rk} is calculated, a random number as noise in the interval $[-maxN, maxN]$ is also calculated and used to modify Δc_{rk} to $\Delta c'_{rk} = \max\{0, \Delta c_{rk} + noise\}$. The parameter $maxN$ is defined as $\eta \cdot \max_{u,v \in V}\{d_{uv}\}$, where η is a control parameter. Also the decision if the noise should be used is adapted using the same idea presented in Section 2.4.4.

The ALNS heuristic can be applied to solve PDPTW with homogeneous and heterogeneous vehicle fleet. For the homogeneous case, the objective is primarily to minimize the total used vehicles and then the total costs. Thus, a two-phase method is used to solve the PDPTW with homogeneous fleet. The first stage is called the *vehicle minimization* stage and works as follows: First of all, an initial feasible solution in which all requests are served is constructed. After that, one route is removed from this feasible solution. The requests in the removed route are put into a request bank and each of them is given a very high penalty cost. An LNS heuristic without adaptive mechanism is then used to solve this problem. The high penalty costs make the heuristic try to insert all requests in the bank into vehicle routes. As long as a feasible solution is found, one new route is removed from the solution and the process repeats. If the LNS heuristic fails to find a feasible solution after a predefined number of iterations and more than 5 (inclusive) requests are unplanned in the request bank, this stage terminates. Otherwise, the whole stage terminates when a maximum iteration number is reached. The last feasible solution is used as the initial solution for the second stage, in which the heuristic is used normally to minimize the total costs.

3 Freight Consolidation

In this chapter, the second type of the basic decision problems in the operational transportation planning of freight forwarding companies is discussed. Besides the possibility of disposing vehicles for the fulfillment of customer requests, forwarders can also transfer some requests to common carriers. In this case, the forwarders do not have to create any routing plan for the common carriers but just pay the freight charges calculated for every O-D pair based on predefined tariff structure. In order to minimize the total freight charges for LTL requests, forwarders can take advantage of economy of scale through consolidating their requests before releasing them to common carriers. The problem of bundling LTL requests to minimize the total charges is called the freight consolidation problem (FCP).

This chapter is organized as follows. Section 3.1 gives an introduction on the FCP. A formal definition of the FCP and a mixed integer programming (MIP) model of this problem are presented in Section 3.2. In order to solve the FCP, two heuristic approaches are presented in Section 3.3. Results of the computational experiments are reported in Section 3.4. Finally, this chapter is concluded in Section 3.5.

3.1 Introduction

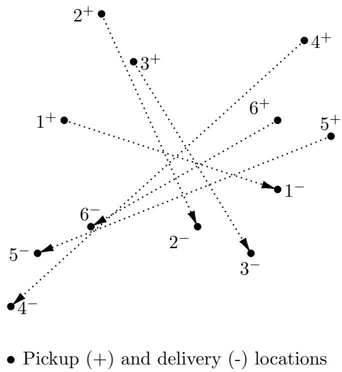
Transferring some requests to common carriers and just pay for the transportation services bought is a very practical strategy to acquire additional flexible capacities for the short-term. This option may be very interesting for freight forwarders if some requests can hardly be consolidated with other ones to construct efficient routes or simply in case the total capacities of the own disposable vehicles are not enough.

If some requests are outsourced to common carriers, the responsibility of the transportation of these requests is also transferred to them. LTL requests can be consolidated with other requests that the common carriers have acquired from their own customers and served by the common carriers' vehicles. Since the concrete routing plan is unknown to the forwarder, a payment for the transportation service based on characters of the routes in which the outsourced requests are fulfilled is not valid. Instead, a more general payment scheme is used. Quotations for the services are proposed based on the common carriers' fixed tariff tables under nonlinear consideration of distance, weight and type of

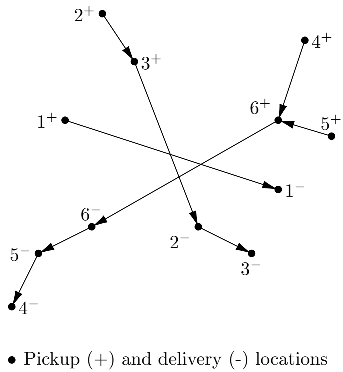
X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_3,

goods specified in the tasks (Ballou, 1999; Kopfer, 1992). Generally, the more loads are to be transported as well as the longer the mileage is, the less the average transportation cost per ton per kilometer is. This characteristic of such tariff structures reflects the economy of scale in transportation.

The objective of the FCP is to construct least-cost flows for the freight calculation so that the total charges paid to the common carriers are minimized. In the FCP, flows of goods of a request can be delivered to locations of other requests and consolidated with other requests over common arcs. The freight charge is calculated for each single arc used in the FCP solution. It is worth mentioning that the result of the FCP does not imply any specification for the execution of a bundle of requests but only provides a basis for determining the total freight charges.



(a) Requests without bundling



(b) Bundles of requests

Figure 3.1: *An example of freight consolidation*

Figure 3.1 shows an example of consolidating LTL requests studied in the FCP. In Figure 3.1a, six LTL pickup and delivery requests are depicted. A result of the consolidation is given in Figure 3.1b. Three bundles are constructed in this result: $\{1\}$, $\{2,3\}$, and $\{4,5,6\}$. Request 1 can hardly be consolidated with other requests and represents a single-request bundle. The freight charge for its fulfillment remains the same before and after the consolidation process. This charge is calculated as the charge for the arc $(1^+,1^-)$ as:

$$\gamma_1 = c_{1^+1^-} = f(q_{1^+1^-}, d_{1^+1^-}) = f(\ell_1, d_{1^+1^-})$$

$c_{uv} = f(q_{uv}, d_{uv})$ is the cost function used for the freight calculation which captures the degressive character of the tariff table. $q_{1^+1^-}$ is the total load transported over the arc

$(1^+, 1^-)$, $d_{1^+1^-}$ is the distance of this arc, and ℓ_1 is the load of request 1. Requests 2 and 3 share both a pickup region and a delivery region and can be well consolidated. Before the freight consolidation, the two requests are separately charged as:

$$\gamma_2 = c_{2^+2^-} = f(q_{2^+2^-}, d_{2^+2^-}) = f(\ell_2, d_{2^+2^-})$$
$$\gamma_3 = c_{3^+3^-} = f(q_{3^+3^-}, d_{3^+3^-}) = f(\ell_3, d_{3^+3^-})$$

After consolidation, three arcs $(2^+, 3^+)$, $(3^+, 2-)$, and $(2^-, 3^-)$ are used and three charges are calculated as:

$$\begin{aligned}\gamma_2 + \gamma_3 &= c_{2^+3^+} + c_{3^+2^-} + c_{2^-3^-} \\ &= f(q_{2^+2^-}, d_{2^+2^-}) + f(q_{3^+2^-}, d_{r_3^+ r_2^-}) + f(q_{3^+2^-}, d_{3^+2^-}) \\ &= f(\ell_2, d_{2^+3^+}) + f(\ell_2 + \ell_3, d_{3^+2^-}) + f(\ell_3, d_{2^-3^-})\end{aligned}$$

Suppose that $\ell_2 = \ell_3$ and $d_{2^+2^-} = d_{3^+3^-}$, and without consolidation these two requests will be charged the same amount of 280 monetary unit (MU) based on a cost rate ψ_1 per ton per kilometer which is dependent on ℓ and d. After consolidation, a cost rate ψ_2 higher than ψ_1 will be applied for the arcs $(2^+, 3^+)$ and $(2^-, 3^-)$, and each of these two arcs will be charged 100 MU. For the third arc $(3^+, 2^-)$ however, since the load is doubled, a cost rate ψ_3 will be applied which is larger than ψ_1 but far below $2\psi_1$ because of the degressive cost function. This leads to a charge for this arc in the amount of 330 MU. As a result, $280 + 280 - 100 - 100 - 330 = 30$ MU can be reduced through freight consolidation. The case of the third bundle $\{4, 5, 6\}$ is similar.

Early research on the FCP can be found in Kopfer (1990, 1992). An A*-algorithm and a genetic algorithm are proposed to solve the FCP. The scenario studied in these two articles is similar to the situation in the VRP, which means all requests have the same pickup location which is the depot. FCP with transportation requests with different pickup and delivery locations as the case in the PDP is studied as part of the IOTPP in Krajewska and Kopfer (2009). A mathematical formulation of the FCP is proposed in this paper. However, no computational result based on this formulation is reported. In order to solve the complicated IOTPP, a TS heuristic is proposed. Another mathematical formulation only for the FCP is proposed by Struwe et al. (2012). The authors also present an SA to solve this problem and compare the performance of this heuristic with the optimal solutions obtained by using their MIP model.

3.2 Mathematical formulation

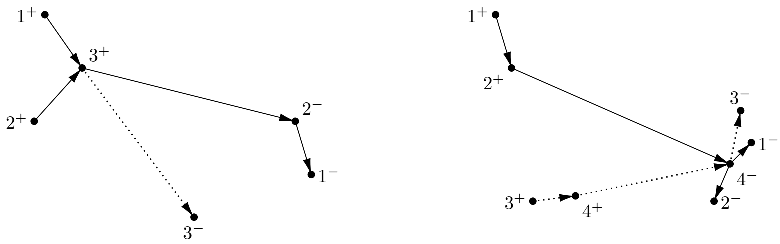
The FCP with LTL pickup and delivery requests can be formulated according to Struwe et al. (2012) in the following way. The FCP can be formally defined on a graph $G = (V, A)$. Let $R = \{1, \ldots, n\}$ denote the set of n requests. $P = \{1, \ldots, n\}$ is the set of pickup locations and $D = \{n+1, \ldots, 2n\}$ is the set of delivery locations. Even when two or more customer nodes refer to the same geographical location, they are represented by different nodes in V. Each request $r \in R$ represents the transportation of an amount of goods ℓ_r from the pickup node $r^+ = r \in P$ to the corresponding delivery node $r^- = r + n \in D$. Define $V = P \cup D$, which is the set of all customer nodes. For each arc $(u, v) \in A = V \times V$, a distance $d_{uv} \geq 0$ and a capacity limit Q_{uv} are given. It is further assumed that the graph is a symmetric one, so that $d_{uv} = d_{vu}$, for all $u, v \in V$. If the amount of loads $q_{uv} \leq Q_{uv}$ is transported over arc (u, v), the freight charge for this freight flow c_{uv} is calculated according to a cost function $c_{uv} = f(q_{uv}, d_{uv})$. Due to the economic of scale, the average cost reduces with increasing degree of resource utilization. We thus have the two degressive properties of this cost function: $c(q_1, d) + c(q_2, d) \geq c(q_1 + q_2, d)$ and $c(q, d_1) + c(q, d_2) \geq c(q, d_1 + d_2)$. The first property indicates that for the same transportation distance, the more loads are transported, the lower the cost rate is. The second one means that for a given load, the longer the distance is, the lower the average cost is.

The task of the FCP is to find a path $p_r = (r, \ldots, r + n)$ for each request $r \in R$, which begins at its pickup location r, over some nodes belonging to other requests, and ends at its delivery location $r+n$ in the way that the total freight charges paid for all used arcs are minimized. If request r is not consolidated with other requests for the freight calculation, its path is $(r, r + n)$. Let A_r be the set of arcs in path p_r. Freight charges c_{uv} will be determined for each arc $(u, v) \in \{(u, v)|(u, v) \in \cup_{r \in R} A_r\}$. For a given arc $(u, v) \in A$, let $R_{uv} \subseteq R$ be the set of requests with arc (u, v) in their paths. c_{uv} will be calculated as $f(\sum_{r \in R_{uv}} \ell_r, d_{uv})$ instead of $\sum_{r \in R_{uv}} f(\ell_r, d_{uv})$, while $\sum_{r \in R_{uv}} \ell_r$ must not exceed the flow capacity Q_{uv} on the arc. For those arcs with $R_{uv} = \varnothing$, $c_{uv} = f(0, d_{uv}) = 0$. The objective of the FCP is to minimize the total freight charges $C = \sum_{(u,v) \in A} c_{uv}$.

The FCP consists of two dependent subproblems. The first one is to divide R into m request bundles with $\cup_{i=1}^{m} B_i = R$ and $B_i \cap B_j = \varnothing, \forall i, j = 1, \ldots, m, i \neq j$. A bundle with only one request is a single-request bundle. A subgraph $G_b = (V_b, A_b)$ of G can be defined for each bundle $b = B_1, \ldots, B_m$. V_b is the customer node set of requests in b, $A_b = V_b \times V_b$. The second subproblem is to find a spanning tree $T_b = \{[u, v]|(u, v) \in \cup_{r \in b} A_r\}$ of undirected edges $[u, v]$ ignoring the directions of the arcs for each bundle b on G_b, $b = B_1, \ldots, B_m$, in such a way that for each request in this bundle a feasible path p_r

is defined and the total freight charges are minimized. Denote T'_b as the set of arcs used in bundle b according to T_b, the total freight charges of request bundle b can be calculated as $C_b = \sum_{(u,v)\in T'_b} c_{uv}$.

It is further to be ensured that none of the nodes in V_b is merely used as a consolidation or a deconsolidation point for other requests, without being consolidated or deconsolidated with those requests. In other words, no customer node is allowed to be used solely as a (de)consolidation point for other requests. Figure 3.2 shows two examples of these undesirable situations that must be forbidden in the FCP.



• Pickup (+) and delivery (-) locations

(a) Consolidation point (3^+)

• Pickup (+) and delivery (-) locations

(b) Deconsolidation point (4^-)

Figure 3.2: *Examples of forbidden (de)consolidation*

In Figure 3.2a, requests 1 and 2 are consolidated at the pickup location of request 3. However, they are not further consolidated with request 3 that actually belongs to the single-request bundle including only request 3 (depicted by the dotted line). In Figure 3.2a, requests 1 and 2 are correctly consolidated but inappropriately deconsolidated at the delivery location of request 4, where the load of request 4 cannot actually be deconsolidated from this bundle $\{1, 2\}$. The dotted line shows the right flows of the second bundle with requests 3 and 4, where at the location 4^- the load of request 4 is deconsolidated with those of request 3.

In order to model the FCP, the graph G can be extended by introducing a fictive depot 0 to $G' = (V', A')$, with $V' = V \cup \{0\}$, $A' = V' \times V'$ and $d_{u0} = d_{0u} = 0$ for all $u \in V$. The path p_r can be extended to a fictive round tour $\theta_r = (0, r, \ldots, r+n, 0)$, which starts from and ends at the fictive depot 0. Let q_{uv} $\forall u, v \in V$ be the loads over arc (u, v). The binary variable x_{uvr} $\forall u, v \in V, r \in R$ will be one if and only if arc (u, v) is part of θ_r. Variable $z_{uv} \in \{0, 1\}$ $\forall u, v \in V$ will be one if and only if arc (u, v) is used in any path. Variable $w_{ur} = 1, \ldots, n,$ $\forall u \in V, r \in R$ is an additional variable used for labeling nodes in tours so that subtours are eliminated. The FCP aims to create n fictive round tours which yield minimal freight charges and can be modeled as follows:

$$\min\ C = \sum_{(u,v)\in A} c_{uv} = \sum_{(u,v)\in A} c(q_{uv}, d_{uv}) \tag{3.1}$$

subject to:

$$x_{0rr} = x_{(r+n)0r} = 1 \quad \forall r \in R \tag{3.2}$$
$$\sum_{v\in V'} x_{0vr} = \sum_{u\in V'} x_{u0r} = 1 \quad \forall r \in R \tag{3.3}$$
$$\sum_{u\in V'} x_{uvr} = \sum_{u\in V'} x_{vur} \quad \forall r \in R, v \in V \tag{3.4}$$
$$w_{ur} - w_{vr} + M \cdot x_{uvr} \leq M - 1 \quad \forall r \in R, u, v \in V, u \neq v \tag{3.5}$$
$$M \cdot z_{uv} \geq \sum_{r\in R} x_{uvr} \quad \forall u, v \in V \tag{3.6}$$
$$\sum_{u\in V} z_{uv} \leq 1 \quad \forall v \in D \tag{3.7}$$
$$\sum_{v\in V} z_{uv} \leq 1 \quad \forall u \in P \tag{3.8}$$
$$q_{uv} = \sum_{r\in R} x_{uvr} \cdot \ell_r \quad \forall u, v \in V \tag{3.9}$$
$$q_{uv} \leq Q_{uv} \cdot z_{uv} \quad \forall u, v \in V \tag{3.10}$$
$$x_{uur} = 0 \quad \forall u \in V', r \in R \tag{3.11}$$
$$z_{uu} = 0 \quad \forall u \in V \tag{3.12}$$
$$x_{uvr} \in \{0, 1\} \quad \forall u, v \in V', r \in R \tag{3.13}$$
$$q_{uv} \geq 0 \quad \forall u, v \in V \tag{3.14}$$
$$z_{uv} \in \{0, 1\} \quad \forall u, v \in V \tag{3.15}$$
$$w_{ur} \geq 0 \quad \forall u \in V, r \in R \tag{3.16}$$

The objective function (3.1) minimizes the total freight charges for all used arcs. Constraints (3.2)-(3.4) imply that θ_r is a round tour which starts and ends with arcs $(0, r)$ and $(r + n, 0)$, respectively. Constraint (3.5) eliminates subtours. Constraints (3.6)-(3.8) limit the number of incoming or outgoing arcs connected with delivery or pickup nodes to exclude pure (de)consolidation points. Constraints (3.9) and (3.10) ensure that the capacity restriction is held. Constraints (3.11) and (3.12) forbid self-cycles.

3.3 Solution methodology

Due to the complex bundling structures, the FCP with pickup and delivery requests is a very difficult combinatorial optimization problem. In this section, two heuristic approaches are presented. The first one is an LS heuristic developed based on the TS heuristic in Krajewska et al. (2008) and Krajewska and Kopfer (2009). The original TS heuristic is used to solve the IOTPP which is the main topic in the next two chapters. This heuristic is adapted to solely solve the FCP by considering only the fulfillment option of common carriers. The second heuristic is an SA algorithm proposed by Struwe et al. (2012), which searches more thoroughly in the neighborhood.

3.3.1 A local search heuristic

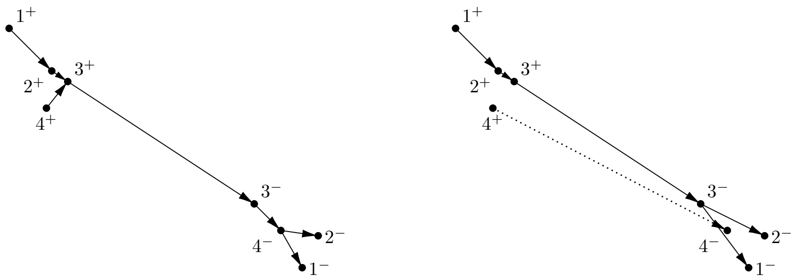
The LS heuristic for the FCP is implemented based on the TS heuristic of Krajewska and Kopfer (2009), in which specific operators are used to deal with the FCP. A more detailed description of the TS heuristic can be found in Krajewska et al. (2008). As it is suggested in Krajewska et al. (2008) that no specific tabu setting is needed for the FCP and the functionality of simultaneously generating vehicle routes is excluded, the original TS heuristic degrades to an LS heuristic. Algorithm 3 shows an outline of the whole heuristic.

Algorithm 3 Local search heuristic

initialize start solution s
$s_{best} := s$
while stopping criterion not met **do**
    randomly choose a request r
    if r belongs to a request bundle with two or more requests **then**
        remove r from the bundle through applying *single-shifting-move* to s
        if s is not feasible **then**
            repair solution s
        end if
        if $f(s) < f(s_{best})$ **then**
            $s_{best} := s$
        end if
    end if
    insert r into s using the *insert-into-flow-move*
    if insertion successful **and** $f(s) < f(s_{best})$ **then**
        $s_{best} := s$
    end if
end while
return s_{best}

The first step is to generate a feasible start solution. The most simple way to do this for the FCP is to build a single-request bundle for each request $r \in R$ without doing any consolidation at all. As a result, the path of each request r is $p_r = (r^+, r^-)$. After the start solution is initiated, the iterative process starts and three operators are used to search the neighborhood: the *single-shifting-move*, the *repair procedure*, and the *insert-into-flow-move*.

The *single-shifting-move* is applied when a request has to be removed from a consolidated request bundle that consists of more than two requests. Figure 3.3 gives an example of this operator. The consolidated bundle of requests 1 to 4 is shown in Figure 3.3a. Given that request 4 has been chosen in line 4 to be removed from the consolidated bundle. The *single-shifting-move* first removes request 4 from the bundle. This is performed by removing the two locations 4^+ and 4^- from all paths containing any of them. In this case, they are the paths of requests 1 and 2, while $p_1 = (1^+, 2^+, 3^+, 3^-, 4^-, 1^-)$ and $p_2 = (2^+, 3^+, 3^-, 4^-, 2^-)$. After removing 4^- from these two paths, they become $p_1 = (1^+, 2^+, 3^+, 3^-, 1^-)$ and $p_2 = (2^+, 3^+, 3^-, 2^-)$. This can be seen in Figure 3.3b. At last, a single-request bundle of request 4 is constructed with its path as $p_4 = (4^+, 4^-)$ (the dotted line in Figure 3.3b).



• Pickup (+) and delivery (-) locations

(a) Request bundle before removal

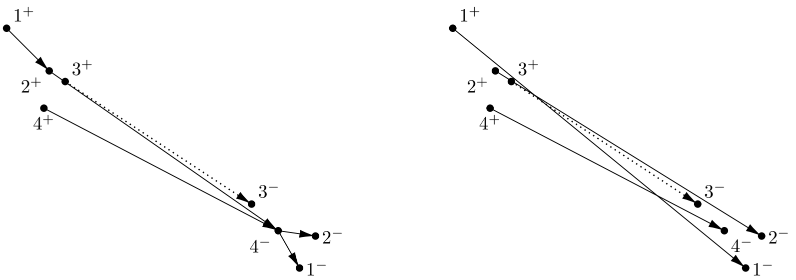
• Pickup (+) and delivery (-) locations

(b) Request bundles after removal

Figure 3.3: *An example of the* single-shifting-move

However, removing requests from bundles using the *single-shifting-move* does not always yield feasible solutions. An example of this failure is shown in Figure 3.4a, when request 3 is chosen to be removed from the bundle $\{1, 2, 3, 4\}$ as shown in Figure 3.3a. After removing the pickup and delivery locations associated with request 3, the paths of the other requests become $p_1 = (1^+, 2^+, 4^-, 1^-)$, $p_2 = (2^+, 4^-, 2^-)$, and $p_4 = (4^+, 4^-)$.

A *repair procedure* described in Algorithm 4 must be applied to the infeasible solution.



• Pickup (+) and delivery (-) locations

(a) Infeasible bundle after removal

• Pickup (+) and delivery (-) locations

(b) Request bundles after repair

Figure 3.4: *An example of the* repair procedure *of the LS heuristic*

Algorithm 4 Repair procedure of the LS heuristic

for all $(u, v) \in T'_b$ **do**
  if $q_{uv} > Q_{uv}$ **then**
    repeat
      remove any request r, $(u, v) \in A_r$ from b
      construct path $p_r = (r^+, r^-)$ for r
      delete r^+, r^- from all remaining paths of requests in b
    until $q_{uv} \leq Q_{uv}$
  end if
end for
for all $r \in b$ **do**
  if $p_r = (r^+, \ldots, l^+, m, \ldots, r^-)$ **and** $p_l \neq (l^+, m, \ldots, l^-)$ **then**
    remove l from b
    construct path $p_r = (r^+, r^-)$ for r
    delete r^+, r^- from all remaining paths of requests in b
  end if
end for
for all $r \in b$ **do**
  if $p_r = (r^+, \ldots, n, l^-, \ldots, r^-)$ **and** $p_l \neq (l^+, \ldots, n, l^-)$ **then**
    remove l from b
    construct path $p_r = (r^+, r^-)$ for r
    delete r^+, r^- from all remaining paths of requests in b
  end if
end for

Suppose that a request has been removed using the *single-shifting-move* from bundle b and b becomes infeasible after the removal. T'_b is the set of arcs used in all requests' paths in b. The *repair procedure* first checks the load situation over all arcs (lines 1-9). Then, the feasibility of consolidation and deconsolidation is checked and repaired when necessary. For the case of Figure 3.3a, this part of the *repair procedure* works in the following way. For request 1, its path is $p_1 = (1^+, 2^+, 4^-, 1^-)$. The first pickup location after 1^+ is 2^+ and 2^+ is followed directly by 4^-. Then, it checks if the path of request 2 has the pattern $(2^+, 4^-, \ldots, 2^-)$ and it is true. Then the procedure goes to the next location in p_1 and gets 4^+ which follows after 2^-. Then, it checks if p_4 is like $(4^+, \ldots, 2^+, 4^-)$ and the answer is false. Thus, the request 1 is removed from the bundle and its new path is $p_1 = (1^+, 1^-)$. The path of request 2 can be repaired in a similar way. As a result, the bundle is completely destroyed after the repair (Figure 3.3b).

The insertion operator works in the following way. Given a request r that should be consolidated with other requests, it first searches for some neighbor nodes that can be used as a consolidation point near the pickup node r^+ and than randomly chooses one of them as the basic point for insertion. Figure 3.5a illustrates an example. Nodes n_1, n_2, and n_3 are the neighbor nodes of r^+, and the node n_3 is chosen as the basic point. The operator then tries to insert the pickup node r^+ in all arcs to which the basic point belongs as well as to construct a direct link to the basic node. In the example of Figure 3.5, these are the three arcs (n_1, n_3), (n_2, n_3), and (n_3, n_4), which are depicted by the thick solid lines in Figure 3.5a. Figures 3.5b to 3.5d show the three possibilities of inserting r^+ into existing arcs and Figure 3.5e illustrates the case of a direct link. The best possibility with the lowest cost increment is then chosen.

The LS heuristic stops after 10,000 iterations in total or after 3,500 iterations without any improvement of the best solution found.

The insertion of the corresponding delivery location functions in a similar manner. Some nodes near the delivery node in the bundle in which the pickup node is inserted are found. One of them is then chosen as basic point. All possible insertions into existing arcs in the bundle with the basic point as an end node as well as a direct link from the basic point to the delivery node are tried. The delivery node is inserted at the best feasible position.

3.3.2 A simulated annealing heuristic

The second algorithm for the FCP is an SA algorithm proposed in Struwe et al. (2012). Solutions to the FCP are represented by m bundles. Since the number of bundles may vary all the time, a dynamic data structure is used. Each bundle b, $b = B_1, \ldots, B_m$, is an array. Each element in the array consists of an arc (u, v) used in the tree of this bundle

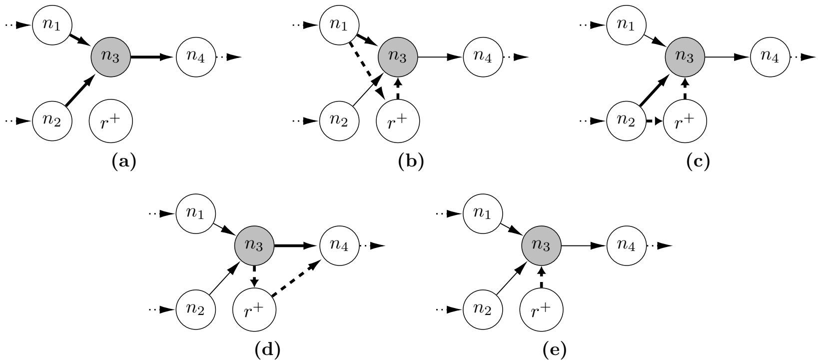


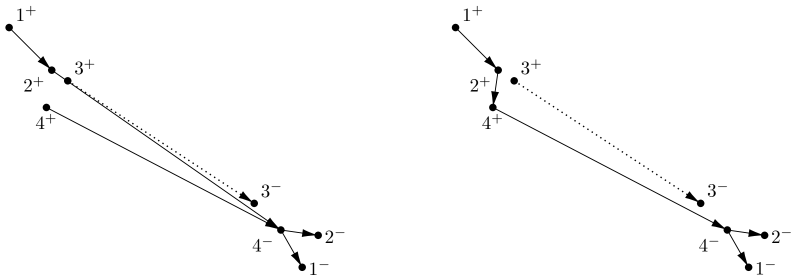
Figure 3.5: *Insertion of a pickup node*

T'_b and a set of requests R_{uv} including all requests transported over this arc, i.e., requests with this arc in their paths.

An initial solution s_0 is obtained by iteratively inserting requests into the solution in random order. In each iteration only one request is inserted and in total, n iterations are needed. The first request is simply inserted as a single-request bundle. In the next iterations, the algorithm evaluates the insertion costs of integrating a request into all existing bundles at the best positions and the cost of constructing a single-request bundle for this request. Let ΔC_b^* be the insertion cost of inserting request $r \in U$ into an existing bundle b at the position that increases the objective value at the least. Let $C_{\{r\}}$ be the insertion cost of creating a single-request bundle for r. U is the set of unplanned requests. The algorithm evaluates ΔC_b^* for all existing bundles and $C_{\{r\}}$ in a parallel fashion and afterward inserts r at the position with the lowest insertion cost.

In each iteration of the SA heuristic, one of two operators is used to generate neighborhood solutions. The *Single-Move* operator is used at a probability of $1 - \beta$. It randomly selects a request, removes it from its current bundle and relocates it at its best position except in its previous bundle. The *Swap* operator, which is used at a probability of β, randomly selects two requests from different bundles and swaps them. If at least one request cannot be inserted into the new bundle, the current iteration is skipped.

As the removal routine works exactly the same as in the LS heuristic described in Section 3.3.1, the problem of having infeasible solutions after the removal (see Figure 3.4a) must be solved, too. A different repair procedure is used in the SA heuristic to fix the problem. The heuristic records during the constructing of each bundle the order in which the requests are inserted. This piece of information is used to repair infeasible solutions. Infeasible bundles will be reconstructed by reinserting the remaining requests into it in the same order as it was constructed before. The motivation of this repair procedure is to preserve more information about the tree structures of the bundles before removal as much as possible. Figure 3.6 illustrates an example of this repair procedure. Suppose that the same request has been removed from the same bundle as shown in Figure 3.4 and that Figure 3.6a shows the same infeasible solution as in Figure 3.4a. Assume that this bundle was constructed in the order 1,2,3,4. Then, after that request 3 is removed, the order of the rest requests is 1,2,4. Figure 3.6b shows the repaired solution after the reconstruction. To diversify the search, the order is shuffled randomly at a small probability of α.



• Pickup (+) and delivery (-) locations

(a) Infeasible bundle after removal

• Pickup (+) and delivery (-) locations

(b) Request bundles after repair

Figure 3.6: *An example of the* repair procedure *of the SA heuristic*

The insertion of a request into a bundle is done in another way as in the LS heuristic. Figure 3.7 shows the examples of inserting the corresponding pickup and delivery locations of a request r. The node with a plus or a minus sign represents a pickup node or a delivery node, respectively. The node with both plus and minus signs can be either a pickup or a delivery node. In order to insert the new request's pickup location r^+ into an existing bundle b, the heuristic checks all possibilities of inserting r^+ into the bundle. There are two types of possible insertion. The first one is to construct a direct link from r^+ to all pickup nodes that already consolidated in the bundle which is illustrated in Figure 3.7a. The second type is to replace every arc used in the bundle $(u, v) \in T'_b$ by two new arcs (u, r^+) and (r^+, v) (Figure 3.7b). After the insertion of the pickup node, the heuristic searches for all feasible insertion options for the delivery location. Again, two types of insertion are possible, i.e., either to construct a direct link from a delivery node after r^+ that exists already in the bundle to r^- (Figure 3.7c), or to replace every arc (u, v) after

r^+ with two new arcs (u, r^-) and (r^-, v) (Figure 3.7d). The case of inserting both the pickup and delivery nodes into an existing arc (Figure 3.7e) is also considered. The best scenario of inserting the request, i.e., the scenario with the lowest cost increment after the insertion of both the pickup and delivery nodes, is realized.

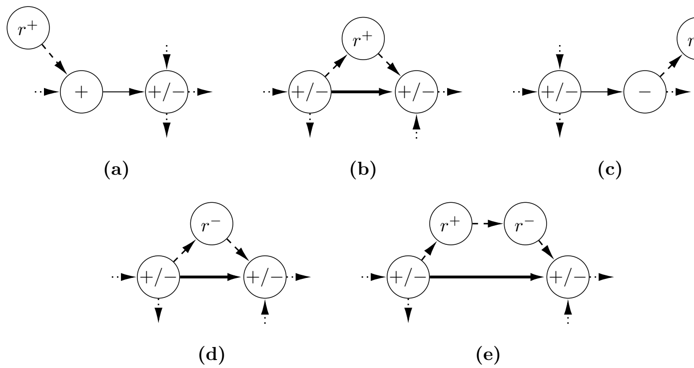


Figure 3.7: *Request insertion*

The SA algorithm uses a geometric cooling schedule that is slightly different as the one presented in Algorithm 2. This algorithm starts with an initial temperature T_0. After n_{rep} iterations in each temperature step, the temperature T is decreased by $\varsigma \times T$, where ς is the cooling factor. The start temperature T_0 is given such a value that a solution that is 15% worse than the initial solution is accepted with probability 0.15. n_{rep} and ς are set to 50 and 0.98 respectively. Both the *Swap* operator and the random shuffle of reinsertion order for bundle reconstruction are used at a probability of $\beta = \alpha = 0.2$. The algorithm stops when a worsening of 1% is accepted at a probability lower than 0.01.

3.4 Computational experiments

The two heuristics presented in the last section are evaluated based on some new theoretical instances. A piece-wise linear cost function for the calculation of freight charges is used so that the results obtained by the heuristics can be compared with the optimal solutions by solving the instances using the formulation presented in Section 3.2.

3.4.1 Cost function for the freight charge calculation

The cost function used to calculate the freight charge for an arc (u, v) is $c_{uv} = \beta^f \eta_{uv} d_{uv}$, where β^f is a basic freight rate per DU and η_{uv} is a price factor that considers the utilization of the vehicle capacity and is a function of the load q_{uv}. The price factor function $\eta = f(q)$ is typically a piecewise linear function that is depicted by the thick solid line in Figure 3.8, where the loads allowed being consolidated over all arcs are limited to 40.

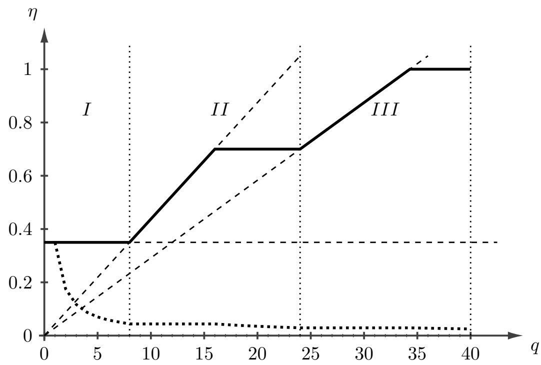


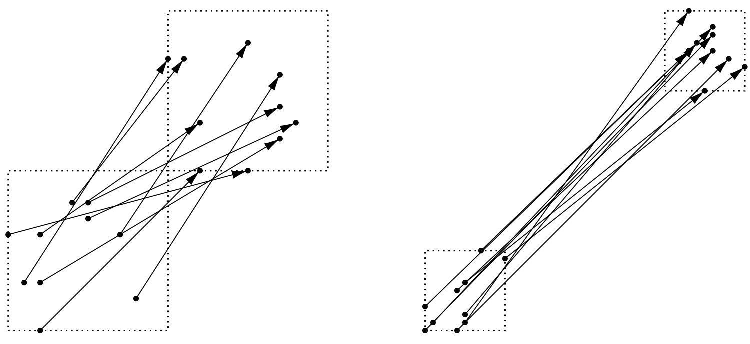
Figure 3.8: *Price factor η as a piecewise linear function of load q*

This piecewise function is derived in the following way. Typically in practice, the freight rates are specified by the common carriers differently for different price ranges. In Figure 3.8, the entire range of load from 0 to 40 are divided by the vertical dotted lines into three ranges that are labeled as I, II, and III. In the price range I, a minimum amount will be charged (the horizontal dashed line). In the other ranges II and III, a fixed cost rate valid in each of them is used so that the cost factor is determined as the amount of load multiplied by the cost rate. The two dashed lines with different slops depict the corresponding cost rate factors. The economies of scale can be seen from the slops of the two dashed lines, i.e., for a higher utilization grade of the capacity (III), a lower rate is applied. However, a direct usage of these two rates would cause the problem that the charge for less loads may be higher than for more loads. For instance, for $q = 24$, which is the division quantity between ranges II and III, the lower factor $\eta = 0.7$ is applied. If the quantity is reduced by one unit to $q = 23$, the higher rate should be used and the factor η would be 0.875 that is greater than 0.7. This implies that the price would be higher for shipping less goods that is quite unfair for the common carriers' customers. In order to solve this problem, the lower factor, i.e., $\eta = 0.7$, is used for the determination of the freight charge. This is the reason for the plateaus in the figure. The actual cost rate factors are depicted by the thick dotted curve. As the cost function is piecewise linear, it can be represented in linear form and thus be solved by the IBM CPLEX solver.

3.4.2 Test instances

For computational analysis, three sets of instances including 20 instances in total are generated. Each instance contains ten pickup and delivery requests. Since the cost-saving effect through freight consolidation is significant when requests have to be picked up from the same region and delivered to another, the instances are generated by firstly defining two disjunctive regions in which only pickup or delivery locations are randomly chosen. The distance between the two regions strongly affects the potential of consolidation. The longer it is, the more cost-savings could be realized. Another important factor that affects the potential of freight consolidation is the distribution of the loads of the requests. The less the load per request is, the more consolidation possibilities there are and the more difficult it is to solve the instance.

Through varying the two factors, i.e., the distance between the pickup and delivery regions and the distribution of the quantity of loads, three sets of instances are generated. In Set 1 (S1), the pickup and delivery locations are randomly determined in a square of (0,0) to (10,10), and in (10,10) to (20,20). The load of a request is randomly set to 0.25 times of the capacity limit Q with a possibility of 20%, to $0.5Q$ with a possibility of 30%, and to $0.75Q$ with a possibility of 20%. Ten instances are generated in S1. In Set 2 (S2), the loads of requests are reduced to a much lower level: 50% requests have a load of $0.1Q$, 30% of $0.2Q$, and 20% of $0.3Q$. In Set 3 (S3), the same load level is used but the distance between the pickup and delivery regions is increased by shifting the delivery region to the square of (30,30) to (40,40). For both Sets S2 and S3, five instances are generated. The two instances 2_01 and 3_02 are shown in Figure 3.9. The pickup and delivery regions are depicted by the rectangles.



(a) Instance 2_01 (short distance) **(b)** Instance 3_02 (long distance)

Figure 3.9: *FCP test instances*

3.4.3 Computational results

In order to identify the cost-saving potential of freight consolidation and to evaluate the efficiency of the two heuristics presented in Section 3.3, the 20 new theoretical instances are firstly solved to optimality using IBM CPLEX 12.5 on an Intel Core i7-2600 computer with 8 cores à 3.40 GHz. The results are given in Table 3.1. The total charges TC' in the second column are obtained using the cost function without performing consolidation. The total charges TC^* in the third column are the optimal solutions found by CPLEX. The next column $\phi = 100 \cdot (TC' - TC^*)/TC'(\%)$ shows how much the cost-saving potential is. All instances are also solved ten times by the two heuristics on a computer equipped with an Intel Core 2 Quad Q6600 CPU. The fifth to seventh columns and the eighth to tenth columns in Table 3.1 report the results. The values of $\Delta = 100 \cdot (TC - TC^*)/TC^*(\%)$ show how much the heuristic results deviate from the optimal solutions. The computational time t is reported in seconds (s).

Table 3.1: *Computational results of the FCP instance*

Inst.		**CPLEX**		**LS**			**SA**		
No.	TC'	TC^*	$\phi^*(\%)$	TC_1	$\Delta_1(\%)$	t_1	TC_2	$\Delta_2(\%)$	t_2
1_01	97.81	77.71	20.55	77.71	0.00	11.2	77.71	0.00	0.6
1_02	94.84	77.19	18.61	77.50	0.40	11.1	78.58	1.80	0.7
1_03	82.45	66.07	19.87	67.56	2.26	11.9	67.81	2.63	0.7
1_04	80.34	68.48	14.76	68.69	0.31	11.5	70.67	3.20	0.8
1_05	87.25	73.84	15.37	74.62	1.06	11.8	76.10	3.06	0.8
1_06	103.46	90.80	12.24	93.53	3.06	11.0	93.58	3.06	0.5
1_07	67.21	57.41	14.57	58.34	1.62	11.6	58.92	2.63	0.7
1_08	81.42	62.74	22.94	64.05	2.09	11.7	65.25	4.00	0.7
1_09	90.12	77.27	14.26	78.26	1.28	11.8	78.34	1.38	0.6
1_10	96.53	83.22	13.79	84.55	1.60	11.8	84.55	1.60	0.5
avg.S1	-	-	16.70	-	1.37	11.5	-	2.34	0.7
2_01	56.00	37.78	32.54	40.18	6.35	11.8	42.42	12.28	1.4
2_02	55.88	32.15	42.46	33.09	2.92	11.9	35.99	11.94	1.8
2_03	54.27	33.96	37.43	35.11	3.38	12.0	37.28	9.78	2.1
2_04	62.77	36.55	41.78	40.54	10.92	11.8	42.20	15.46	1.7
2_05	54.07	37.26	31.09	38.92	4.46	11.7	41.41	11.14	1.5
avg.S2	-	-	37.06	-	5.61	11.8	-	12.12	1.7
3_01	266.62	201.55	24.41	201.55	0.00	11.7	201.55	0.00	0.8
3_02	249.27	185.71	25.50	190.84	2.76	11.7	188.50	1.50	0.7
3_03	228.87	168.09	26.56	173.60	3.28	11.6	169.99	1.13	0.7
3_04	241.19	179.49	25.58	182.67	1.77	11.6	182.67	1.77	0.6
3_05	265.20	199.19	24.89	199.47	0.14	11.6	200.80	0.81	0.8
avg.S3	-	-	25.39	-	1.59	11.6	-	1.04	0.7

The reduction of freight charges ϕ^* shown in the fourth column shows that considerable cost-savings can be achieved through consolidation. Especially for small loads (S2) and long transportation distances between the pickup and delivery regions (S3), freight charges can be reduced considerably up to 42%. This result indicates the practical importance of the FCP if the goods share the pickup and delivery regions. On the contrary, when the pickup and delivery nodes are scattered in the same region and the goods are generally to be transported in different directions, the cost-saving potential of consolidation will no more be significant.

Through a comparison between the two heuristic approaches, it can be concluded that the SA heuristic is much faster than the LS heuristic and performs slightly better for S3. However, for the other sets, especially for S2 that is much more difficult to solve due to the small loads and much more possibilities of the consolidation, the LS obviously outperforms the SA in terms of solution quality.

3.5 Conclusions

In this chapter, the basic decision problem concerning outsourcing LTL requests to external common carriers is discussed. The freight charges that must be paid to the common carriers for the fulfillment of the outsourced requests are determined independently for each direct transport between an origin and a destination. Due to the economics of scale, the longer the distance between the O-D pair is, or the more the loads to be transported are, the cheaper the freight rate in general is. Thus, for the same situations, for instance, when loads are to be transported from pickup locations in a region to different delivery nodes in another region, the total charges can be reduced considerably through freight consolidation for the freight charge calculation. In this context, freight consolidation does not mean to plan any specific routing plans but to calculate the total charges to be paid for the fulfillment of the requests.

The FCP for LTL pickup and delivery requests has been formally defined and modeled in this chapter. The FCP is an extremely complex combinatorial optimization problem. Similar to the PDP, two closely related subproblems must be solved, i.e., the clustering problem that assigns each request to a cluster (in the PDP each cluster means a vehicle), and the problem of connecting the requests in each cluster. In the PDP, the form of the connections within a cluster is a path when the start and end depots of the vehicle are located at different places or a circle when the two depots refer to the same node. It means that each customer node is reached from exactly one node and left for another node exactly once, too. In the FCP, however, the structure of the connections is a tree instead of a chain, and there exist a large number of possible shapes of the tree that have to be examined. In order to solve this complex problem, two heuristic approaches are presented and tested on some new theoretical instances. Computational results confirm the efficiency of the approaches and indicate the importance of this problem for the studied situation, where several requests share the same pickup and delivery regions.

4 From Cherry-Picking to Integrated Operational Transportation Planning

The increasing pressure on modern freight forwarding companies to improve profitability have strongly affected their fleet management and transportation planning strategies for the fulfillment of customer transportation requests. In order to reduce the overall operational costs and to maintain a very high utilization grade of the own transportation resources, freight forwarders need not only optimize their internal processes, but also consider external capacities from other carriers in their operational planning.

In this chapter, the combination of using both own capacities and those from subcontractors, i.e., self-fulfillment and subcontracting is discussed. Utilizing external capacities and integrating the dispatching of these resources into the operational transportation planning enable freight forwarders to reduce their own fleet size. Due to the high fixed costs of vehicles, many freight forwarders strongly reduce the capacity of the own fleet and hold an available capacity far under the varying demands of requests. The gap between the own fleet capacity and customer demands is filled by hiring extraneous capacities from other carriers on different terms that will be discussed in more detail in Section 4.1.

A widely used strategy in practice is the so-called "cherry-picking", i.e., assigning only the most profitable tours to the own vehicles. This two-phase sequential strategy can assure a very high efficiency and profitability of the own fleet, but not necessarily yields execution plans with minimized overall costs. This strategy will be discussed in Section 4.2.

In order to reduce the total fulfillment costs of all customer requests, including both the costs for the own vehicle fleet and the costs paid for the subcontracting, integrated planning that simultaneously disposing own and subcontractors' vehicles has to be performed. This means that the planning of outsourcing requests to subcontractors has to be integrated into the traditional vehicle routing and scheduling. The resulted problem is referred to as the IOTPP and will be discussed in more detail in Section 4.3.

The last section in this chapter will show how the integration of the flexible external capacities that are generally more expensive than the own vehicles can reduce the overall operational costs over a long-term through a simple simulation based on the VRP, when customer demands fluctuate during the planning horizon.

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_4,

4.1 External transportation resources in subcontracting

External transportation capacities can be classified into different *fulfillment modes* according to their different contingents. Compared with the own vehicles, the external capacities provide freight forwarders with much more flexibility of transportation resources in spite of the higher cost rates.

The most discussed fulfillment mode in literature (see Section 4.3.1 for a more comprehensive review on the literature) is to use independent common carriers. Both LTL requests and bundles of them can be fulfilled in this way. The motivation behind this fulfillment mode is that neither these (bundles of) requests can be profitably planned into available vehicle routes nor the limited capacities are sufficient for their execution. However, to decide how much a forwarder has to pay for a specific task consisting of either a single request or a bundle of requests as a whole may be complicated. Generally, there are two pricing possibilities for this fulfillment mode. The first one is based on spontaneous quotation or negotiation between the seller (common carrier) and the buyer (forwarder) of the transportation service. The price could be very flexible because the quotation is strongly influenced by the subjective situation rather than by any established tariff structure. Sometimes, this may lead to very cheap prices. For example, in the empty balancing planning, some vehicles have to be shifted from one depot to another without loading. The planner would like to find out if there exist some tasks which could be acquired in the market and executed by empty, returning vehicles, when these tasks could be accomplished without having any negative impacts on their regular operations in the next planning period.

However, most payments actually happening are not in this manner. Quotations are proposed by subcontractors based on internal fixed tariff tables under non-linear consideration of distance, weight and type of goods specified in the tasks (Ballou, 1999; Kopfer, 1992). Generally, the more loads are to be transported as well as the longer the distance to travel is, the less the average transportation cost per ton per kilometer is. This characteristic of such tariff structures reflects the economies of scale in transportation and it is always preferable to consolidate transportation orders before releasing them to common carriers. If transportation requests are consolidated for the freight calculation, the FCP presented in Chapter 3 has to be solved for this fulfillment mode. Otherwise, freight charges will be decided for each request singly.

Besides the option of common carriers, several other fulfillment modes of subcontracting complete vehicle routes to subcontractors are identified by Kopfer et al. (2006) based on an analysis of a German mid-sized freight forwarder. This kind of subcontracting can be divided into two fulfillment modes, which are applied to subcontractors who are frequently engaged by the forwarder and are nearly exclusively employed by him. These subcontractors usually have signed long-term contracts with the forwarder and can provide up to an agreed number of vehicles that can be planned in a similar way as the own vehicles. To these subcontractors, complete vehicle routes starting from and ending at the forwarder's depot are constructed and outsourced. For the execution of a route, a vehicle of a subcontractor will be paid either *on a tour basis* or *on a daily basis*. The payment on a tour basis is calculated by multiplying the route length with an agreed tariff rate per DU, which is higher than the corresponding cost rate of an own vehicle to compensate the fixed costs. Paying on a daily basis means that a predefined flat-rate is paid to a subcontractor for a complete day, without violating the agreed limits for travel distance of the route. Costs arise only if a vehicle is used at all.

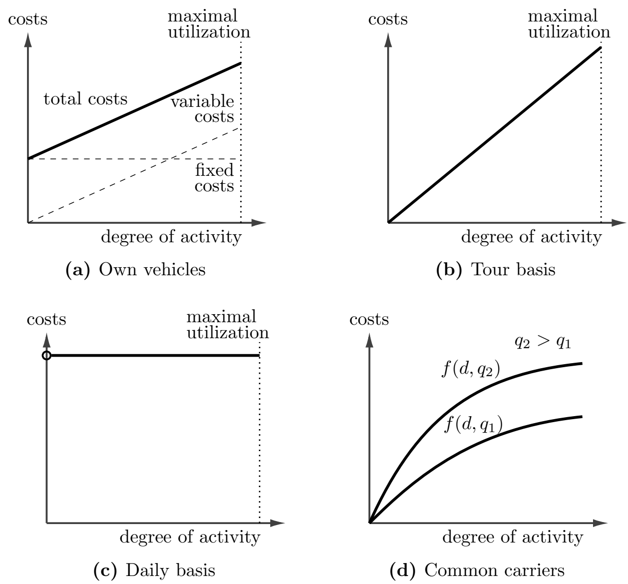


Figure 4.1: *Fulfillment costs of different modes, cf. Krajewska (2008)*

The different cost structures of self-fulfillment and the three modes of subcontracting are illustrated in Figure 4.1. Compared to the own fleet, no fixed costs have to be paid directly in the charge of a vehicle hired on a tour basis. However, the different slops of the two cost lines in Figures 4.1a and 4.1b indicate that the variable cost rate of subcontracting on a tour basis is significantly higher than that of the own vehicles. The flat-rate for the usage of a vehicle hired on a daily basis is generally very high even if the daily limit is exhausted. Thus, this mode is only of interest for requests that do not fit into vehicle routes, for which costs are calculated on a distance basis. The tours of this mode are usually long and run into directions where no favorable bundling is possible (Krajewska, 2008). The curves in Figures 4.1d represent an approximation of the total costs calculated based on the step-wise function for the freight calculation shown in Figure 3.8.

4.2 Cherry-Picking

"Cherry-picking" is a widely used planning strategy by freight forwarding companies in practice and functions in a two-phase manner. In the first phase, requests that can be very well bundled are selected out of the entire request portfolio and integrated into very efficient routes that are reserved for vehicles in the own fleet, so that the own vehicles can result in a very high financial performance. In the second phase, the rest share of the request portfolio is planned with vehicles from subcontractors on a tour and a daily basis, while any left ones are then shifted to common carriers. For the sake of simplicity, the situation where both pickup and delivery locations of customer requests are scattered in the same region is considered. Thus, it is only needed to consider the case that requests are outsourced to common carrier singly, i.e., without being consolidated for the freight calculation discussed in the context of the FCP.

"Cherry-picking" can be formally described as follows. Suppose that a freight forwarder has acquired a request portfolio $R = \{1, 2, \ldots, n\}$ with n pickup and delivery requests. The customer payment for the fulfillment of request r is denoted as ρ_r. Define the set of pickup nodes as $P = \{1, 2, \ldots, n\}$ and the set of delivery nodes as $D = \{n+1, n+2, \ldots, 2n\}$. Thus, the corresponding delivery node of a pickup node $u \in P$ is $u + n$. Let ℓ_r denote the quantity of goods to be transported of request r and $\ell_u = \ell_r$ and $\ell_{u+n} = \ell_r$, for all $u = r$, $r \in R$. An own fleet K^o with ϱ^o vehicles is stationed at the depot of the forwarder o. We use o to denote the start depot and o' the end depot which is a duplicate of o and define the node set as $V = P \cup Q \cup \{o, o'\}$ and the arc set as $A = V \times V$. Operation at each node $u \in V$ must be started within a predefined time window $[b_u, e_u]$ and the service time needed for the operation is s_u. Specifically, the service time at the depot nodes is set to 0. Each vehicle can be loaded up to a capacity limit Q_k, $k \in K^o$. The fixed cost of vehicle k is α_k and the variable cost rate is β_k. The optimization problem in the first phase can be defined then on a graph $G = (V, A)$. The distance of an arc $(u, v) \in A$ is given by d_{uv} and a vehicle needs d'_{uv} time unit to travel over this arc. In order to model this problem, the decision variables as used in the PDPTW model in Section 2.2 are needed. x_{uvk} indicates if vehicle k travels from node u to v. Variables t_{uk} and l_{uk} give the start time of the service at node u and the load of vehicle k after the service at u. A new binary variable z_r^s is introduced here to indicate if a request is selected for self-fulfillment in this phase. Since $R = P$, index u can be used instead of r in ρ_r and z_r^s. The objective in the first phase is to maximize the profit realized by using only the own vehicles and the problem can be formulated as follows:

$$\max \sum_{u \in P} \rho_u z_u^s - \sum_{k \in K^o} \alpha_k - \sum_{k \in K^o} \sum_{(u,v) \in A} \beta_k d_{uv} x_{uvk} \tag{4.1}$$

subject to:

$$\sum_{k \in K^o} \sum_{v \in V \setminus \{o\}} x_{uvk} = z_u^s \quad \forall u \in P \tag{4.2}$$

$$\sum_{v \in P \cup D} x_{uvk} - \sum_{v \in P \cup D} x_{v,n+u,k} = 0 \quad \forall k \in K^o, u \in P \tag{4.3}$$

$$\sum_{v \in P \cup \{o'\}} x_{ovk} = 1 \quad \forall k \in K^o \tag{4.4}$$

$$\sum_{u \in D \cup \{o\}} x_{uo'k} = 1 \quad \forall k \in K^o \tag{4.5}$$

$$\sum_{u \in V \setminus \{o'\}} x_{uvk} - \sum_{u \in V \setminus \{o\}} x_{vuk} = 0 \quad \forall k \in K^o, v \in P \cup D \tag{4.6}$$

$$x_{uvk}(t_{uk} + s_u + d'_{uv} - t_{vk}) \leq 0 \quad \forall k \in K^o, (u,v) \in A \tag{4.7}$$

$$b_u \leq t_{uk} \leq e_u \quad \forall k \in K^o, u \in V \tag{4.8}$$

$$t_{uk} + d'_{u,n+u,k} \leq t_{n+u,k} \quad \forall k \in K^o, u \in P \tag{4.9}$$

$$x_{uvk}(l_{uk} + \ell_v - l_{vk}) = 0 \quad \forall k \in K^o, (u,v) \in A \tag{4.10}$$

$$\ell_u \leq l_{uk} \leq Q_k \quad \forall k \in K^o, u \in P \tag{4.11}$$

$$0 \leq l_{n+u,k} \leq Q_k - \ell_u \quad \forall k \in K^o, u \in P \tag{4.12}$$

$$l_{ok} = 0 \quad k \in K^o \tag{4.13}$$

$$x_{uvk} \in \{0,1\} \quad \forall k \in K^o, (u,v) \in A \tag{4.14}$$

$$z_u^s \in \{0,1\} \quad \forall u \in P \tag{4.15}$$

$$t_{uk} \geq 0 \quad \forall k \in K, u \in V \tag{4.16}$$

$$l_{uk} \geq 0 \quad \forall k \in K, u \in V \tag{4.17}$$

Assume that the set of requests that are not assigned in the first phase is $R^{sub} \subset R$. Define the graph $G^{sub} = (V^{sub}, A^{sub})$ in analogy to G according to R^{sub}. Let K^t and K^d denote the sets of the hired vehicles on a tour basis and on a daily basis, respectively. For the vehicles hired on a tour basis, a cost rate $\beta_k, k \in K^t$ is applied, which is higher than that for the own vehicles. For a used vehicle that is hired on a daily basis, the flat-rate is

α_k and its maximal route length is L_k, $k \in K^d$. Suppose that the charge of outsourcing a request $r \in R^{sub}$ to a common carrier is γ_r. The binary variable z_k^d, $k \in K^d$ indicates if a vehicle paid on a daily basis is used. The binary variable z_r^c, $r \in R^{sub}$ indicates if request r should be given to a common carrier. Again, the index u is used instead of r as well as P^{sub} instead of R^{sub}. The planning problem in the second phase can be formulated as a cost minimization problem for the customer payments will be held by the forwarder:

$$\min \sum_{k \in K^t} \sum_{(u,v) \in A^{sub}} \beta_k d_{uv} x_{uvk} + \sum_{k \in K^d} \alpha_k z_k^d + \sum_{u \in P^{sub}} \gamma_u z_u^c \tag{4.18}$$

Constraint (4.2) has to be reformulated as:

$$\sum_{k \in K^t \cup K^d} \sum_{v \in V^{sub}} x_{uvk} + z_u^c = 1 \quad \forall u \in P^{sub} \tag{4.19}$$

and the following constraints related to vehicles paid on a daily basis must be added into the model:

$$\sum_{v \in P^{sub}} x_{ovk} = z_k^d \quad \forall k \in K^d \tag{4.20}$$

$$\sum_{(u,v) \in A^{sub}} d_{uv} x_{uvk} \leq L_k \quad \forall k \in K^d \tag{4.21}$$

Equation (4.20) ensures that if vehicle $k \in K^d$ goes to some customer node, it must be paid by forcing $z_k^d = 1$. Constraint (4.21) restricts the maximal length for these vehicles.

4.3 Integrated operational transportation planning

Although "cherry-picking" can help achieve a very high profitability of the own vehicles, it may result in very bad routes for the subcontracting modes and lead to inferior solutions compared to a simultaneous minimization of the total execution costs. In order to solve the IOTPP to achieve the minimum total costs, integrated planning approaches that simultaneously dispose all fulfillment modes discussed in the previous section are needed.

The IOTPP considering all possible fulfillment modes, i.e., self-fulfillment, subcontracting on a tour and a daily basis as well as subcontracting to common carriers, consists of three interdependent subproblems as shown in Figure 4.2. The first one is the mode-selection problem, which assigns requests to different fulfillment modes. The second subproblem is the vehicle routing and scheduling problem, which has to be solved for both the own fleet and the vehicles of subcontractors paid on a tour and a daily basis. The third subproblem is the FCP for requests to be outsourced to common carriers. It is worth mentioning that in case of full-truckload (FTL) or where the LTL requests can hardly be consolidated for the freight charge calculation, the FCP can be omitted.

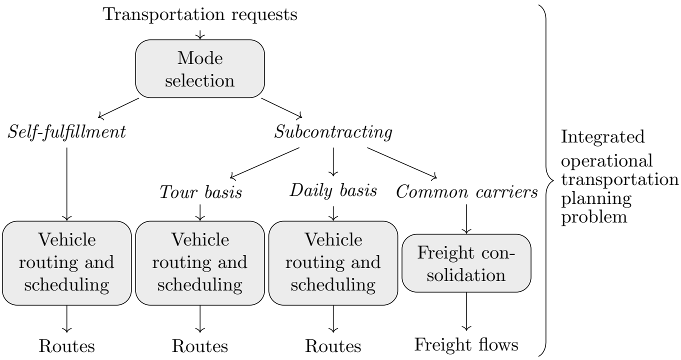


Figure 4.2: *Structure of the IOTPP*

4.3.1 Literature review

Due to the variety of fulfillment modes and fee charge terms, the integrated planning of different combinations of internal and external transportation capacities has been discussed in the last decades. With little exceptions, the research on IOTP can be classified in three streams. The first stream of research on the integrated planning is to consider the option of shifting complete vehicle routes to subcontractors. The second stream focuses on the combination of self-fulfillment and subcontracting to common carriers. The third one refers to the integrated planning taking both subcontracting complete routes and subcontracting requests that are charged due to freight tariffs into account.

4.3.1.1 Subcontracting vehicle routes

The problem of simultaneously routing vehicles in the own fleet and those from the subcontractors can be seen as a special case of the problem of routing a heterogeneous fleet of vehicles. Literature reviews on the heterogeneous fleet vehicle routing problem (HVRP) can be found in Baldacci et al. (2008) and Hoff et al. (2010).

In the HVRP, several different vehicle types are available for the request fulfillment. The number of available vehicles of type g is given by ϱ_g. Each vehicle of the same type g has a fixed capacity Q_g, a fixed cost α_g, and a variable cost rate β_g. It is commonly assumed that the fixed cost of a vehicle with larger capacity is also higher than a smaller one.

Baldacci et al. (2008) classify the HVRP into several variants. The criteria considered in the classification are (1) whether the fleet size is limited, (2) whether the fixed costs of vehicles are considered, and (3) whether the routing costs are identical for all vehicles. In the fixed fleet version of the HVRP, which is usually referred to as a *heterogeneous* routing problem, the number of vehicles ϱ_g is limited for each vehicle type g. This problem focuses on the best usage of the given fleet and thus is generally solved on the operational level. The unlimited fleet size version of the HVRP is usually named with “fleet size and mix” and solved on a higher, i.e., a tactical or even a strategic level, to get the optimal fleet composition.

In the context of the integrated transportation planning, the fixed costs of the own vehicles, which can also be heterogeneous, do not need to be considered on the operational level but on the tactical or strategic level. It it worth mentioning that the heterogeneity of the vehicle fleet in the integrated transportation planning problems is not only a result of different sizes of the vehicles, e.g. in Li et al. (2007) and Brandão (2009), but also of the different fee charge models applied to the subcontracting modes discussed in Section 4.1, even when the capacity of all vehicles is the same. This fact makes the problem more complicated since the cost effectiveness of different types of subcontractors’ vehicles depends also on the utilization of the vehicles, and thus, none of the subcontracting modes should be generally preferred as the large vehicles in case of the typical HVRP.

Savelsbergh and Sol (1998) study a problem in which some vehicles are rented permanently that can be regarded as own vehicles and others are rented according to the total amount of requests. A column generation based algorithm is proposed to solve the problem. In Strumpf (1998) it is assumed that vehicles can be rented in the short term if they are needed and the number of vehicles that can be hired from subcontractors is variable. The maintenance costs of these vehicles are covered partly by the fees charged for the working time of vehicles and drivers. Some heuristic approaches are proposed to solve this problem.

4.3.1.2 Subcontracting requests to common carriers

The second main research stream focuses on another extension of the routing problems in such a way that not all requests need to be served by vehicle routes, similar to the problems discussed in e.g. Butt and Ryan (1999) and Feillet et al. (2005). The requests that cannot be served using the own fleet are transferred to common carriers in form of either single request or as bundles of requests that have been consolidated by solving the FCP. The freight charge paid for the outsourced requests to common carriers can be modeled as a penalty cost in the objective function.

Ball et al. (1983) investigate the problem of simultaneously generating routes for the own leased vehicles and subcontracting requests singly, which are represented as O-D pairs, to common carriers to minimize the total fulfillment costs. The underlying routing problem can be seen as a multi-depot PDP for FTL requests. The strategic problem of determining the size of the leased fleet is also discussed by solving the problem using aggregated data of a chemical firm. Two route-first cluster-second approaches and a greedy insertion heuristic are presented. After the routing problem is solved, all routes are ordered by decreasing route savings that is defined as the cost difference between using an own vehicle and the total charges in case the common carrier option is used. Requests in the least-saving routes are outsourced to common carriers.

Klincewicz et al. (1990) study a fleet size problem on the strategic level under the consideration of using common carriers in the situation where a single warehouse is serving a geographic area with random daily customer demands. In order to solve this stochastic problem, the whole area is divided into a set of sectors. Each of these sectors is then assigned either to the mode self-fulfillment or to a common carrier. The problem is then modeled as a single-source facility location problem and solved with a Lagrangian relaxation heuristic. The results suggest reasonable fleet sizes for the long-run.

A static and deterministic version of this problem on the operational level is proposed and solved in Chu (2005) that was published several years later. In this problem, only a fixed number of heterogeneous trucks with limited capacity are available in the own fleet and the demands of customers are known. Requests can be either fulfilled by self-fulfillment or by subcontracting to common carriers while the FCP is not considered. A heuristic approach called TL-LTL is proposed to solve this problem, which firstly assigns the requests with the lowest outsourcing costs to common carriers and generates routes for the rest share of the requests for the own vehicles just like a normal VRP.

The same problem and its variant with a homogeneous fleet have been investigated in Bolduc et al. (2007), Bolduc et al. (2008), and Côté and Potvin (2009). Bolduc et al. (2007) also follow a sequential solution strategy as the one used in Chu (2005) but apply a more complected neighborhood structure in the LS heuristic used to improve the routes. In Bolduc et al. (2008), however, an additional operator that enables a reassignment of requests between the self-fulfillment and subcontracting is introduced and it leads to significant better results. Côté and Potvin (2009) refer to the problem as the VRP with private fleet and common carrier and develop a TS heuristic. The option of common carrier is represented as a fictive route so that the heuristic can also search for better solutions by mode exchange with conventional inter-route exchange operators used for the VRP.

4.3.1.3 Subcontracting both routes and requests

Discussions on two more comprehensive variants of the IOTPP that consider both the options of shifting complete vehicle routes to subcontractors and of transferring requests to common carriers without constructing any concrete routes can be found in Krajewska (2008) and Krajewska and Kopfer (2009), as well as in Ceschia et al. (2011), respectively.

An extension of the PDPTW considering all subcontracting modes introduced previously in Section 4.1, i.e., subcontracting on a tour and a daily basis as well as the employment of common carriers, is studied in Krajewska (2008) and Krajewska and Kopfer (2009). Specifically for the mode common carriers, the authors also consider the FCP to further reduce the total freight charged for the requests outsourced to common carriers. In order to solve this problem, a TS heuristic is proposed that simultaneously considers all fulfillment modes to achieve good quality solutions. Besides the operators for the FCP that have been presented in Section 3.3.1, an *insert-into-vehicle-route* operator and a *swapping-routes* operator are used to construct vehicle routes and to improve the solution by switching the fulfillment modes of two vehicle routes. The computational results based on real-world data show considerable cost-savings achievable through the IOTP.

Ceschia et al. (2011) study an extension of the VRP by subcontracting of different modes. The requests are divided into two subsets. The mandatory orders must be fulfilled by some vehicles while for the optional orders certain penalty costs are paid if they are not delivered. The introduction of the penalty cost of an optional order can be regarded as the option of subcontracting it to the common carrier while the freight charges are specified for each single request individually. In other words, the FCP is not considered. Four payment models for employing subcontractors' vehicles are introduced: (1) a fixed cost plus a variable cost calculated as the product of a cost rate per DU and the length of the routes, (2) a fixed cost plus a variable cost calculated as the product of a cost rate per weight unit and the load of the request that lies at the farthest location in the route, (3) a mixture of models (1) and (2) in dependence of a threshold load value, and (4) a fixed cost plus a variable cost which depends both on the total load and on the farthest location. A TS is developed to solve this problem and used to solve both this new problem and the one studied in Chu (2005). Computational results show the effectiveness of this

4.3.2 Mathematical model

In this section, a mathematical model is presented for the IOTPP studied in this thesis. The four fulfillment modes introduced in Section 4.1 are all considered. For the sake of simplicity, only the case that requests are outsourced singly without freight consolidation is considered for the mode common carriers.

Since the customer payments are constant, the IOTPP intends to minimize the overall fulfillment costs for all four modes simultaneously to maximize the total profits. Using the same notations defined in Section 4.2, the IOTPP can be modeled in the following way, where $K = K^o \cup K^t \cup K^d$ is the entire available fleet:

$$\min C = \sum_{k \in K^o} \alpha_k + \sum_{k \in K^o \cup K^t} \sum_{(u,v) \in A} \beta_k d_{uv} x_{uvk} + \sum_{k \in K^d} \alpha_k z_k^d + \sum_{u \in P} \gamma_u z_u^c \tag{4.22}$$

subject to:

$$\sum_{k \in K} \sum_{v \in V \setminus \{o\}} x_{uvk} + z_u^c = 1 \quad \forall u \in P \tag{4.23}$$

$$\sum_{v \in P \cup D} x_{uvk} - \sum_{v \in P \cup D} x_{v,n+u,k} = 0 \quad \forall k \in K, u \in P \tag{4.24}$$

$$\sum_{v \in P \cup \{o'\}} x_{ovk} = 1 \quad \forall k \in K \tag{4.25}$$

$$\sum_{u \in D \cup \{o\}} x_{uo'k} = 1 \quad \forall k \in K \tag{4.26}$$

$$\sum_{u \in V \setminus \{o'\}} x_{uvk} - \sum_{u \in V \setminus \{o\}} x_{vuk} = 0 \quad \forall k \in K, v \in P \cup D \tag{4.27}$$

$$x_{uvk}(t_{uk} + s_u + d'_{uv} - t_{vk}) \leq 0 \quad \forall k \in K, (u,v) \in A \tag{4.28}$$

$$b_u \leq t_{uk} \leq e_u \quad \forall k \in K, u \in V \tag{4.29}$$

$$t_{uk} + d'_{u,n+u,k} \leq t_{n+u,k} \quad \forall k \in K, u \in P \tag{4.30}$$

$$x_{uvk}(l_{uk} + \ell_v - l_{vk}) = 0 \quad \forall k \in K, (u,v) \in A \tag{4.31}$$

$$\ell_u \leq l_{uk} \leq Q_k \quad \forall k \in K, u \in P \tag{4.32}$$

$$0 \leq l_{n+u,k} \leq Q_k - \ell_u \quad \forall k \in K, u \in P \tag{4.33}$$

$$l_{ok} = 0 \quad k \in K \tag{4.34}$$

$$\sum_{v \in P} x_{uvk} = z_k^d \quad \forall k \in K^d \tag{4.35}$$

$$\sum_{(u,v) \in A} d_{uv} x_{uvk} \leq L_k \quad \forall k \in K^d \tag{4.36}$$

$$x_{uvk} \in \{0,1\} \quad \forall k \in K, (u,v) \in A \tag{4.37}$$

$$z_u^c \in \{0,1\} \quad \forall u \in P \tag{4.38}$$

$$z_k^d \in \{0,1\} \quad \forall k \in K^d \tag{4.39}$$

$$t_{uk} \geq 0 \quad \forall k \in K, u \in V \tag{4.40}$$

$$l_{uk} \geq 0 \quad \forall k \in K, u \in V \tag{4.41}$$

4.4 Reducing long-term costs through integrated planning — A computational example based on the VRP

Through employing external capacities of subcontractors, freight forwarding companies can improve the efficiency of the own fleet and reduce their long-term costs in the transportation market with fluctuating customer demands. Although the external capacities are generally more expensive than the own fleet, they offer a higher flexibility. A simple simulation study conducted in Kopfer and Wang (2009) illustrates the benefits of the downsizing that many freight forwarders do in practice.

The study of Kopfer and Wang (2009) is based on the VRP. In the IOTP variant of the VRP which is referred to as the vehicle routing and forwarding problem (VRFP), all four fulfillment modes introduced in Section 4.1 are considered. The results of a five-day planning of two compositions of the fleet are compared. In the VRP, there are 5 identical vehicles in the own fleet that can cover the peak of the customer demands. In the VRFP, a mixed fleet with two own vehicles, one vehicle hired on a tour basis, and one vehicle hired on a daily basis is used. Every day, a random number of customers from 9 to 12 are to be served. The demands of the customers are generated according to the Poisson distribution $q \sim Poisson(8)$ while the capacity of the vehicle is restricted to 25 for all vehicles. The fixed cost associated with an own vehicle is 500 and the variable cost per DU is 0.8. For each driven DU of the vehicle hired on a tour basis 1.7 MU have to be paid and the flat-rate for the employment of the vehicle hired on a daily basis is 630. The maximal tour length of both the own vehicles and those hired on a tour basis is set to 850 and for the vehicle paid on a daily basis to 400. For the flow based subcontracting option, the freight charge is determined for each single request shifted to the common carriers individually according to a fixed rate per DU of 3. The VRP and the VRFP are both formulated as MIP models and solved by CPLEX to optimality. Figure 4.3 illustrates the optimal solution for instance D1. Table 4.1 shows the results of the experiment.

Table 4.1: *Comparison between VRP and VRFP at the operational level*

	VRP					**VRFP**		
Ins.	VC	FC	TC	#Vcl.	η_1(%)	TC	#Vcl.	η_2(%)
D1	2153.6	2500.0	4653.6	5	100	4228.8	2	100
D2	1491.2	2500.0	3991.2	4	80	2847.7	2	100
D3	1704.0	2500.0	4204.0	4	80	3360.6	2	100
D4	1482.4	2500.0	3982.4	5	100	2880.7	2	100
D5	1734.4	2500.0	4234.4	4	80	3325.5	2	100
Sum	8565.6	12500.0	21065.6	-	-	16643.3	-	-

In case of the VRP, the forwarder has to hold an own fleet with at least 5 vehicles so that he can fulfill all customer requests on his own. The second to the fourth columns in

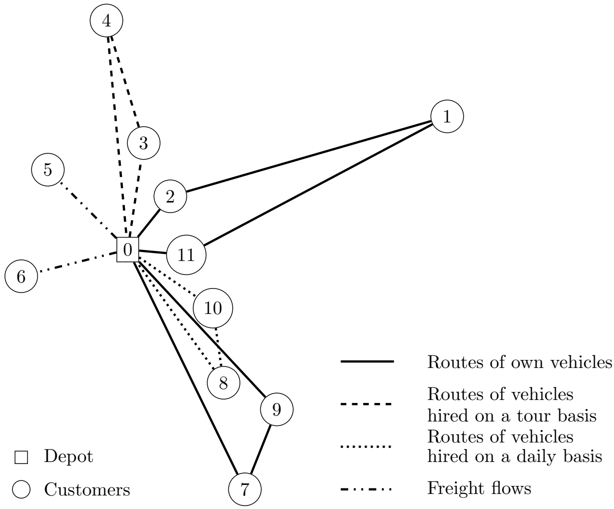


Figure 4.3: *Solution of a VRFP instance*

Table 4.1 show the variable costs (VC), the fixed costs (FC), and the total costs (TC) of the results. The fifth column gives the number of actually used vehicles on these days (#Vcl.). Although not all the five vehicles are needed every day, the same amount of the fixed costs still have to be paid. The numbers in the sixth column (η_1) indicate how efficient the own fleet is used. Due to the standstill of one vehicle on the three days D2, D3, and D5 in the simulation, the fleet is only 88% utilized on average.

On the contrary, the integrated planning VRFP enables the forwarder to do downsizing of the own fleet (from 5 to 2 vehicles) without necessarily rejecting any customers. Even when the customer demands change from day to day, the two own vehicles in the test setting are very efficiently used as indicated in the last column (η_2). The benefits of incorporating the flexible capacities from subcontractors can be seen through the comparison of the total fulfillment costs in the last row. The total costs of the VRFP solutions account to a reduction of more than 20% against the VRP solutions. For the long-term planning at a higher level, it is an essential issue for the forwarder to determine the own fleet size as well as the number of vehicles from subcontractors he will need to obtain by signing contracts on different terms. In order to study this issue, Kopfer and Wang (2009) extend the simulation and solve the VRFP without fixing the number of available vehicles. Table 4.2 summarizes the outputs.

The optimal total costs and the actual utilization of the vehicles of the operational

Table 4.2: *Comparison between operational and strategic planning*

	Operational planning		**Lower bound for strategic planning**	
Ins.	TC	Used vehicles	TC	Opt. fleet
D1	4228.8	2/1/1	4167.2	2/2/1
D2	2847.7	2/1/0	2847.7	2/1/0
D3	3360.6	2/1/1	3360.6	2/1/1
D4	2880.7	2/1/0	2848.7	1/2/0
D5	3325.5	2/1/0	3325.5	2/1/0
Sum	16643.3	-	16549.7	-

planning under the consideration of the predefined fleet composition, i.e., two own vehicles and in each case one vehicle hired either on a tour or a daily basis, are given in columns two and three. The data in column three are given in the format $\varrho^o/\varrho^r/\varrho^d$, where ϱ^o, ϱ^r, and ϱ^d are the numbers of vehicles in the own fleet, hired on a tour basis, and hired on a daily basis, respectively.

For the fleet size problem, the optimal fleet composition is determined for each single instance and given in column five in the same format as in column three. The summed total costs of all five instances given in the last row represent a lower bound for the best possible case since it is assumed that the fleet composition could be specified for each single day. However, the choice of the predefined fleet composition appears to be reasonable for the test setting since the resulted total costs are only 0.57% higher than the lower bound.

It is further to be mentioned that this simulation assumes deterministic data sets for each single day, which is usually not available for the tactical and strategic decision-making. For the long-term planning, some aggregated information should be gathered. The problem instances solved in this simple simulation is very small. The practical instances to be solved every day by freight forwarders are much larger and thus cannot be solved using standard solver like CPLEX. Instead, heuristic approaches that can find sub-optimal solutions within acceptable time limits have to be developed. In the next chapter, two heuristic approaches are presented for this challenging problem.

5 Solution approaches for the integrated operational transportation planning problem

The IOTPP introduced in Chapter 4 for LTL pickup and delivery requests with time windows extends the classical PDPTW by different modes of subcontracting and represents a complicated combinatorial optimization problem. Specifically, the heterogeneity of the vehicles in the IOTPP attributes mainly to the different cost structures of the fulfillment modes rather than to the physical specifications of the trucks. Compared to the PDPTW with homogeneous fleet, the solution space is usually smaller when additional restrictions are considered in a heterogeneous counterpart due to different specifications of the vehicles. A very common example is the transportation of fresh foods that requires refrigerated trucks. On the contrary, when the same route can be fulfilled by vehicles that are equally equipped but charged on different modes, the solution space of the problem with heterogeneous fleet will be larger than the case of a homogeneous PDPTW due to varying cost structures.

Theoretically, both types of heterogeneity can be considered in the IOTPP. However, for the sake of simplicity, the vehicles in the IOTPP considered in this thesis are only different due to their payments. An exception is associated with the vehicles hired on a daily basis, which are additionally given a maximum route length to avoid extreme long routes for this mode. Moreover, all customer nodes are scattered in the same region, so that the FCP is not specifically considered as a subproblem of the IOTPP modeled in Section 4.3.

In order to solve the IOTPP, two heuristic approaches proposed in Wang et al. (2014) are presented in this chapter. The first one is a further development of the ALNS proposed by Ropke and Pisinger (2006) for the PDPTW that is described in detail in Section 2.4. Through some modifications that are made for the specific problem structure of the integrated planning, this ALNS heuristic is also able to solve the IOTPP. The needed modifications are described in Section 5.1. The second one presented in Section 5.2 is developed by modeling the IOTPP as a set partitioning problem (SPP) or a set covering problem (SCP). It applies the ALNS heuristic in an iterative fashion to search for promising partial solutions and finally recombines them into better solutions. The idea of this approach is similar to the *column generation* technique that has been successfully used to solve the routing problems. Both heuristics are tested on new generated theoretical

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_5,

instances and the computational results are reported in Section 5.3. The whole chapter is then concluded in Section 5.4.

5.1 Adaptive large neighborhood search

As introduced previously in Chapter 2, the ALNS heuristic proposed by Ropke and Pisinger (2006) is one of the most promising heuristic in literature for solving the PDP. The heuristic is constructed using an SA framework, while in each iteration one removal and one insertion operator are chosen and used. The probability of applying a removal or an insertion operator is adapted during the search process. A more detailed description of this heuristic can be found in Section 2.4.

In order to make the heuristic suitable for the IOTPP, two major modifications are made to the original ALNS heuristic. The first modification handles the problem whether to insert a request into an existing vehicle route when it is possible. The second modification deals with the specific cost structure of the fulfillment mode *subcontracting on a daily basis*.

In a normal PDPTW, all requests must be planned in vehicle routes so that all requests will be inserted by the insertion operators of the ALNS heuristic so long as it is possible. In the IOTPP, however, it is not the case as requests that are not planned in any route are shifted to common carriers. Thus, a decision must be made when a request r can be inserted into some route k but may result in a higher insertion cost Δc_{rk} than the freight charge γ_r for outsourcing it to the common carriers. Simply rejecting any insertion that is “more expensive” than the common carrier charge will prohibit the use of any vehicles at the beginning phase of route construction because the insertion costs are usually high. Especially when the very first request is inserted into an empty route, the insertion cost can be much higher than the freight charge due to the empty miles. On the other hand, if the more expenses of inserting requests into routes compared to the freight charges are tolerated too much, the solution quality can also be diluted. Thus, a balanced strategy is needed to deal with this problem.

Different as the insertion operators used in the original ALNS heuristic (see Section 2.4.3), which calculates for each request r the insertion cost Δc_{rk}, the *cost savings* $\gamma_r - \Delta c_{rk}$ that can be achieved by inserting request r into route k are calculated by the modified ALNS insertion operators. Here, cost savings refer to the amount of cost that can be reduced if a request is inserted into a route instead of being outsourced to a common carrier. Requests with positive cost savings are automatically considered as candidates for insertion later on. But also requests with negative cost savings are given the chance to be inserted into routes if $\Delta c_{rk} - c_r \leq \zeta T_{it}$ holds, where ζ is a threshold parameter and

T_{it} is the temperature of the SA in iteration it. At the beginning of the search process, as long as T_{it} is large, the heuristic tries to insert all requests into vehicle routes. Thus, the resulted vehicle routes are generally quite efficient. In later phases, T_{it} becomes much smaller and the heuristic becomes more and more selective. Requests will only be (re-)inserted into vehicle routes if the insertion reduces the overall objective value (at least at the moment when the insertion is done).

While inserting requests into vehicle routes, another modification has to be made for the mode subcontracting on a daily basis. The cost function used for calculating the insertion cost of a request into a vehicle of this mode is changed, since otherwise the insertion cost for any request in an empty vehicle route would be the flat-rate, and after that, the insertion cost would be zero for all other requests. A fictive variable cost rate is thus calculated for this mode by dividing the flat-rate by a route length that is slightly shorter than the maximal agreed route length. The insertion cost is then determined as the product of this fictive cost rate and the increment of the length of the route.

5.2 Heuristic II: An iterative approach

In this section, a sophisticated approach, given by *HII*, is proposed. In general, *HII* is similar to the column generation technique used for solving vehicle routing problems. The IOTPP is modeled as an SPP and the decision problem is to choose a set of candidate vehicle routes of different fulfillment modes as well as to specify the requests to be outsourced to common carriers in such a way that the total fulfillment costs are minimized. In order to construct the candidate routes, heuristic *HII* uses the ALNS heuristic iteratively to find efficient vehicle routes and recombines them to generate IOTPP solutions.

5.2.1 A set partitioning model of the IOTPP

The IOTPP can be modeled as an SPP in the following way. Let Ω^o denote the set of all feasible routes that can be executed by a vehicle in the own fleet K^o. Similarly, Ω^t and Ω^d represent the sets of feasible routes that can be executed by vehicles hired either on a tour basis or on a daily basis, respectively. The numbers of routes in these sets are denoted as ω^o, ω^t, and ω^d, respectively. $\Omega = \Omega^o \cup \Omega^t \cup \Omega^d$ represents all feasible routes for all three fulfillment modes, and the total number of routes in the united set Ω is given by $\omega = \omega^o + \omega^t + \omega^d$.

For each route $j \in \Omega$, c_j represents the route costs. For the own vehicles, only the variable costs are considered. For vehicles hired on a tour basis, c_j is calculated by multiplying the cost rate with the length of the routes. For vehicles hired on a daily basis, c_j is the fixed daily tariff flat-rate. If a route can be executed by vehicles of more than one mode, it will be represented as different routes in Ω, since the expense of using different modes to fulfill this route may be different. The binary variables f_j^o, f_j^t, and f_j^d are used to specify the mode of a route j. For instance, $f_j^o = 1$ means that route j is executed by an own vehicle. Each vehicle route j is associated with exactly one fulfillment mode, which means the equation $f_j^o + f_j^t + f_j^d = 1$ holds for all $j \in \Omega$.

A vehicle route may serve several customer requests. The coefficient $a_{rj} = 1$ indicates that a request r is served in route j, and $a_{rj} = 0$ otherwise. The binary decision variable y_j, $j \in \Omega$ indicates if a route is chosen for request fulfillment. Furthermore, we still use γ_r to denote the freight charge that must be paid for outsourcing request $r \in R$ to a common carrier and let $z_r^c = 1$ indicate this. The IOTPP can be defined as to choose a set of vehicle routes for the request fulfillment and outsource the rest share of the request portfolio to common carriers in such a way that the total fulfillment costs are minimized. This decision problem, which is the so-called *master problem*, can be modeled as follows:

$$\min C = \sum_{k \in K^o} \alpha_k + \sum_{j \in \Omega} c_j y_j + \sum_{r \in R} \gamma_r z_r^c \tag{5.1}$$

subject to:

$$\sum_{j \in \Omega} a_{rj} y_j + z_r^c = 1 \quad \forall r \in R \tag{5.2}$$

$$\sum_{j \in \Omega} f_j^o y_j \leq \omega^o \tag{5.3}$$

$$\sum_{j \in \Omega} f_j^t y_j \leq \omega^t \tag{5.4}$$

$$\sum_{j \in \Omega} f_j^d y_j \leq \omega^d \tag{5.5}$$

$$y_j \in \{0, 1\} \quad \forall j \in \Omega \tag{5.6}$$

The objective function (5.1) minimizes the total request fulfillment costs including the fixed costs of the own fleet, the total costs of vehicle routes, and the charges for outsourcing requests singly to common carriers. Constraint (5.2) ensures that every request must be executed by exactly one route or alternatively be outsourced to a common carrier. The following three constraints are the fleet size restrictions. The hired vehicles will only be paid if they are utilized. For the own vehicles, however, the fixed costs are included in the total costs, even in case that some of them are not used, which means that the left-hand side of constraint (5.3) is strictly smaller than the right-hand side.

5.2.2 Construction of candidate routes

The problem of finding feasible vehicle routes is defined as the *subproblem*. Since the number of all possible vehicle routes increases exponentially with the problem size, it is impossible to enumerate all of them. Practically, only a subset of $\Omega' \subset \Omega$ will actually be generated. To improve the solution quality obtained by using Ω', an iterative process is applied, in which only promising routes that can improve the objective function value are searched and added into Ω'. This procedure is similar to the column generation technique for vehicle routing and can be summarized in the following way.

The very first step of *HII* is to generate a meaningful Ω' with sufficient vehicle routes that can be used to initiate the iterative process. This can be done by calling the ALNS heuristic to solve the IOTPP and recording the best solutions found by the heuristic during the whole search process. All vehicle routes in these recorded solutions are put into Ω', while duplicates are considered only once.

Given a sufficiently meaningful subset Ω' of Ω and the decision variables y_j correspondingly, the procedure iteratively solves the LP-relaxation of the master problem by replacing (5.6) with:

$$y_j \geq 0 \quad \forall j \in \Omega' \tag{5.7}$$

and adds new routes/variables into the model when needed just like in the *simplex method*, which looks in each iteration for a promising variable to enter the basis so that the objective function value can be improved. An iteration in column generation consists (1) of solving the LP-relaxation of the master problem in order to determine the current optimal objective function value and the dual multipliers, and (2) of finding, if there still exists one, a route j with negative reduced cost that can be added into Ω'. Denote the dual variables associated with restriction (5.2) as $\boldsymbol{\pi}$ and with restrictions (5.3)-(5.5) as σ^o, σ^t, and σ^d, respectively, the reduced cost $\bar{c_j}$ of a column j is defined as follows:

$$\bar{c_j} = c_j - \sum_{r \in R} \pi_r a_{rj} - \sigma^o f_j^o - \sigma^t f_j^t - \sigma^d f_j^d \tag{5.8}$$

A common way to solve the subproblem of finding a route with negative reduced cost $\bar{c_j} < 0$ is to solve the problem of minimizing the reduced cost $\bar{c_j}$ subject to that the route found by solving the subproblem is feasible. The objective function of this problem can be formulated in the following way:

$$\min c_j - \sum_{r \in R} \pi_r a_{rj} - \sigma^o f_j^o - \sigma^t f_j^t - \sigma^d f_j^d \tag{5.9}$$

The best result of this optimization problem is denoted as $\bar{c_j^*}$. If $\bar{c_j^*} \geq 0$, there exists no variable that can be entered into the basis and improve the LP-relaxed master problem and the iteration is terminated. Because each vehicle route is assiciated with exactly one fulfillment mode, only one term from the last three in (5.9) is actually relevant for the given mode of a route. Thus, for a heterogeneous vehicle fleet as in the IOTPP, this subproblem can alternatively be solved for each fulfillment mode individually. If the optimal solution satisfies the integrality requirement (5.6), the optimal solution to the LP-relaxed master problem is also the optimal solution to the original problem. Otherwise, the solution offers only a lower bound for the IOTPP and additional algorithms have to be applied to get an integer solution.

Different from solving the minimization problem to get the route with the smallest reduced costs, *HII* uses two options to find a set of complimentary routes at a time, even if some of them may have non-negative reduced costs.

Option one solves the IOTPP for the available entire vehicle fleet K using the ALNS heuristic presented in Section 5.1 with the objective of minimizing the total reduced costs of all vehicle routes in the IOTPP solution. This can also be understood as to find a giant tour for the entire fleet K that is to be split into several routes. Through using the index k for vehicles in K instead of j for routes, the objective function of the subproblem solved in *HII* can be formulated as follows:

$$\min \sum_{k\in K} \bar{c_k} = \sum_{k\in K} c_k - \sum_{k\in K}\sum_{r\in R} \pi_r a_{rj} - \sum_{k\in K} \sigma^o f_k^o - \sum_{k\in K} \sigma^t f_k^t - \sum_{k\in K} \sigma^d f_k^d \tag{5.10}$$

The first term on the right-hand side calculates the total costs of routes in the solution without the fixed cost of own vehicles. The second term sums up the π_r values for those requests chosen for the vehicle fleet and can be rewritten as $\sum_{r\in R}(1 - z_r^c)\pi_r$, where the binary variable $z_r^c = 1$, $r \in R$ indicates that request r is outsourced to a common carrier and $z_r^c = 0$ otherwise. As the remaining terms $\sum_{k\in K} \sigma^o f_k^o$, $\sum_{k\in K} \sigma^t f_k^t$, and $\sum_{k\in K} \sigma^d f_k^d$ are constants, the objective function (5.10) is equivalent to:

$$\min \sum_{k\in K} c_k - \sum_{r\in R} \pi_r(1 - z_r^c) = \sum_{k\in K} c_k + \sum_{r\in r} \pi_r z_r^c - \sum_{r\in R} \pi_r \tag{5.11}$$

The constant term $\sum_{r=1}^{n} \pi_r$ can be omitted again and this objective function (5.11) can be reformulated by substituting $\sum_{k\in K} c_k$ in a more concrete form and by adding the constant term calculating the fixed costs of own vehicles:

$$\min \sum_{k\in K^o} \alpha_k + \sum_{k\in K^o\cup K^t} \sum_{(u,v)\in A} \beta_k d_{uv} x_{uvk} + \sum_{k\in K^d} \alpha_k z_k^d + \sum_{r\in R} \pi_r z_r^c \tag{5.12}$$

The binary decision variable z_k^d, $k \in K^d$ indicates whether a vehicle of the mode subcontracting on a daily basis is actually used. This objective function (5.12) is actually the same as (4.22), except that the charges for outsourcing requests to common carriers are replaced by the dual variable values $\boldsymbol{\pi}$. That is why we can directly use the ALNS heuristic to generate new routes. Since the IOTPP with heterogeneous fleet is solved with (5.12), the heuristic *HII* using this option is denoted as *HII-HET*.

Instead of solving the IOTPP for the entire fleet K, the second option solves several subproblems for every vehicle mode, i.e., for the vehicle sets K^o, K^t, and K^d, respectively. The corresponding objective functions are given by (5.13)-(5.15). As a result, each IOTPP instance solved here has a homogeneous and smaller fleet. This variant of the heuristic *HII* can thus be denoted as *HII-HOM*.

$$\min \sum_{k \in K^o} \alpha_k + \sum_{k \in K^o} \sum_{(u,v) \in A} \beta_k d_{uv} x_{uvk} + \sum_{r \in R} \pi_r z_r^c \tag{5.13}$$

$$\min \sum_{k \in K^t} \sum_{(u,v) \in A} \beta_k d_{uv} x_{uvk} + \sum_{r \in R} \pi_r z_r^c \tag{5.14}$$

$$\min \sum_{k \in K^d} \alpha_k z_k^d + \sum_{r \in R} \pi_r z_r^c \tag{5.15}$$

Both *HII-HET* and *HII-HOM* solve the corresponding subproblems in a heuristic manner by applying the ALNS heuristic. A specific advantage of using a heuristic approach as the route generator to solve the subproblem is that a number of different high-quality solutions can be obtained at a time. Each time the ALNS heuristic is used, up to φ best solutions found during the search process are recorded. Candidate routes are then derived from the routes in these recorded solutions and subsequently added into Ω'. The iterative process of generating new routes stops when some stopping criterion is satisfied.

The operator that derives candidate routes from the recorded solutions checks for each single route in these solutions whether it is also a feasible route for other fulfillment modes. In this case, this route will be added into Ω' more than once with different route costs calculated for different modes. For instance, a route constructed for a vehicle hired on a daily basis can also be executed by an own vehicle or a vehicle hired on a tour basis. This route will thus be added into Ω' for each of the three modes as different routes. This operation can be understood as an special form of the *swapping-routes move* used in Krajewska (2008) and Krajewska and Kopfer (2009), which changes two vehicle routes of different modes. Computational tests show that the application of this operation leads to considerable improvements of the solution quality.

5.2.3 Obtaining integer solutions

After the route generation process has been terminated, the master problem is solved again by considering all vehicle routes generated in the iterative phase that have been added into Ω' while the constraint (5.2) is replaced by:

$$\sum_{j \in \Omega'} a_{rj} y_j + z_r^c \geq 1 \quad \forall r \in R \tag{5.16}$$

This model, i.e., (5.1), (5.3)-(5.6), and (5.16) is an SCP-based formulation of the master problem. The motivation for choosing the SCP-based formulation for the master problem instead of further using the LP-relaxed SPP-based model is to minimize the efforts needed to get a feasible integer solution to the original IOTPP if the solutions of these two relaxations are not feasible to the SPP-based model of the IOTPP. In classical column generation based solution approaches, the LP-relaxation is further used. Since the integrity of the binary variable y_j has been relaxed, a solution to the relaxed problem will be infeasible to the IOTPP because some variables y_j have fractional values. In order to get a feasible integer solution, a Branch-and-Bound (B&B) search process that may put great demands on the computational efforts has to be applied.

On the contrary, to get a feasible solution to the IOTPP from an SCP solution is much easier. The infeasibility of an SCP solution is resulted by the fact that the relaxation of the SCP-based model allows each request to be assigned more than once. More precisely, some requests may be assigned to several vehicle routes. In this case, the solution to the SCP-based model will be repaired at the end of *HII*. The solution repair routine first removes all requests that have been assigned to several vehicles from the related routes. Then it tries to reinsert them into vehicle routes at the best position using the insertion operators of the ALNS while keeping the remaining part of the result unchanged.

5.2.4 Overview of the heuristic

The heuristic *HII* presented in this section is constructed based on the ALNS heuristic introduced in the previous section 5.1. It is used both for the generation of an initial route set Ω' and for finding new promising routes in the iterative procedure.

The entire heuristic *HII* consists of 6 steps:
Step 1: Solve the original IOTPP using the ALNS heuristic and record up to φ different best solutions found so far. Add all routes derived from the recorded solutions into Ω'.
Step 2: Solve the LP-relaxation of the SPP-based formulation of the master problem. If it is the first iteration, go to Step 4, otherwise go to Step 3.

Step 3: Check whether the objective function value has been improved more than a threshold value ε compared to the result of the last iteration. If true, go to Step 4, otherwise go to Step 6.
Step 4: Substitute γ_r with π_r for all $r \in R$ and add the fixed terms into the IOTPP model. Solve the heterogeneous IOTPP for all fulfillment modes (*HII-HET*) or several homogeneous IOTPP instances for each single fulfillment mode of vehicle routing (*HII-HOM*) with updated values using the ALNS heuristic. Record up to φ best solutions found so far. Add all routes derived from the recorded solutions into Ω'.
Step 5: Check if the given number of maximal iterations has been reached. If true, go to Step 6, otherwise go back to Step 2.
Step 6: Solve the SCP-based formulation of the master problem and repair the solution when necessary.

It is further to mention that in Step 2, the SCP-based formulation of the master problem can also be used by replacing (5.2) with (5.16). Computational results show no difference in performance between the two formulations.

5.3 Computational experiments

In order to evaluate the efficiency of the proposed solution approaches for the IOTPP, new theoretical instances are generated based on the benchmark PDPTW instances generated by Li and Lim (2001), which are widely used in literature. Since it is obvious that the IOTP is superior to "cherry-picking" regarding the overall execution costs, the experiments of "cherry-picking" are omitted.

5.3.1 Instance generation

The new instances generated for the IOTPP are derived from the PDPTW instance set lc100, lr100, and lrc100. In order to convert a PDPTW instance to an IOTPP instance, two tasks have to be completed: (1) to generate a heterogeneous fleet composed of vehicles of different fulfillment modes, and (2) to generate for each request a freight charge in case that it is to be outsourced to a common carrier.

Two questions have to be answered to generate a heterogeneous fleet including all three fulfillment modes of vehicle routing, i.e., self-fulfillment, subcontracting on a tour basis, and subcontracting on a daily basis. The first one is to decide the fleet size, i.e., how many vehicles of each mode there are in the heterogeneous fleet, and the second one is to determine the cost structure for the fleet. The total fleet size ϱ for the IOTPP instances is set to the total number of used vehicles according to the best-known solutions for the PDPTW that can be found on the website of SINTEF[1]. The numbers of vehicles of the three fulfillment modes, ϱ^o, ϱ^t, and ϱ^d count to about 40%, 35%, and 35% of the total vehicle number ϱ, respectively.

Cost structure data can be determined according to the following steps. Firstly, a reference route length L_{ref} which is meanwhile the maximal route length for vehicles of mode subcontracting on a daily basis L_k, $k \in K^d$ is defined. Secondly, a variable cost rate β^t for the mode subcontracting on a tour basis is defined and used for calculating a reference route cost $c^t_{ref} = \beta^t L_{ref}$. The route costs for the other two modes can then be calculated using c^t_{ref}. For the mode subcontracting on a daily basis, the route cost which is the flat-rate for this mode is set to 90% of c^t_{ref}. For the mode self-fulfillment, the route cost is set to about 80% of c^t_{ref} for the same length, while 60% of this amount is calculated as the fixed cost and the other 40% as the variable cost. For the computational experiments conducted here, the reference route length L_{ref} is set to 80, which is about 75% of the average route length of all 29 chosen PDPTW instances (lc100, lr100, and lrc100) that accounts to 104.7. This average route length can be easily calculated using the best-known solutions documented by SINTEF. The basic cost rate β^t is set to 3.8. However, for each instance, we further adjust β^t a little bit by adding a random term (up to 0.38, i.e. 10% of 3.8). Finally, the calculated values are rounded.

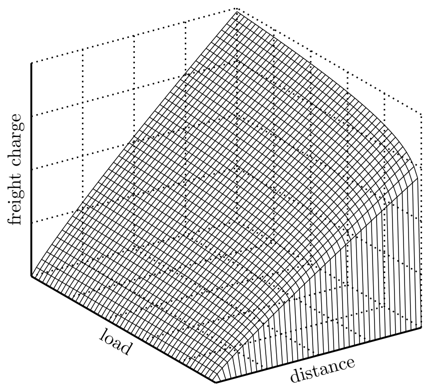


Figure 5.1: *Freight charge function of Krajewska and Kopfer (2009)*

If a request should be forwarded to a common carrier, a freight charge must be paid for the transportation service offered by the common carrier. The amount of the charge depends on the load to be transported and the distance between the pickup and delivery

[1] http://www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark/

locations. Krajewska and Kopfer (2009) suggest an approximation of the freight function for this mode. For a request r with load ℓ_r that must be transported from node u to node v, the charge can be calculated as $\gamma_r = \beta^f \cdot (d_{uv} \cdot \ell_r^{\lambda})^{1-\lambda}$, where β^f is a constant tariff rate and λ is a parameter that takes values between 0 and 1. Figure 5.1 depicts the freight charge function used by Krajewska and Kopfer (2009).

However, the used benchmark instances contain some requests, whose pickup and delivery locations are the same. It results in some requests with transportation distance of zero, which in turn makes the charge of zero according to the freight function of Krajewska and Kopfer (2009). In order to fix this problem, a distance independent term is added into the freight charge function. The new function is $\gamma_r = \beta_1^f \cdot \ell_{cal} + \beta_2^f \cdot (d_{uv} \cdot \ell_r^{\lambda})^{1-\lambda}$, where β_2^f is another constant rate and $\ell_{cal} = max\{\ell_r, \ell_{min}\}$. ℓ_{min} is the minimal load used for the calculation. In this research, the parameters β_1^f, β_2^f, and λ are set to 1.5, 9, and 0.2, respectively. ℓ_{min} is calculated for each instance as 1.2 times of the smallest request load. The adjusted new freight function is shown in Figure 5.2.

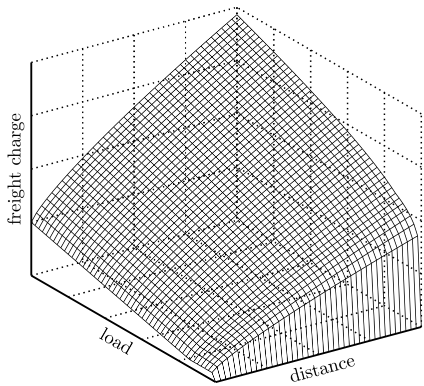


Figure 5.2: *Adjusted freight charge function*

5.3.2 Computational results

The heuristics presented in Sections 5.1 and 5.2 are used for solving the IOTPP. While solving the IOTPP instances using the ALNS heuristic, the same parameter setting for the operators as suggested in Ropke and Pisinger (2006) are used. For the SA process, however, the start temperature is set for each instance individually in such a way that a solution which is 10% worse than the current solution is accepted with probability 0.5. The cooling rate ς is not set fixed for all instances, but determined as such that after Φ

iterations the temperature is reduced to 5% of the start niveau, where Φ is the maximal number of iterations of the SA process. The heuristic stops when $\Phi = 25,000$ or after $\Phi/3$ iterations without any improvement of the best solution found so far.

For Heuristic *HII*, the ALNS heuristic is used iteratively for route generation. For both the two variants *HII-HET* and *HII-HOM*, the ALNS runs up to $\Phi = 10,000$ iterations in the first step. In the following steps, it runs only $\Phi = 2,500$ iterations each time. The whole process is stopped after the iterative route generation process has been repeated five times or after three consecutive abortive attempts to improve the objective function of the LP-relaxed master problem for at least $\varepsilon = 5\%$. Each time, up to $\varphi = 1,000$ best solutions found during the search process are recorded. The routes derived from these solutions are added into the route set Ω'. If a route can be executed by vehicles of different modes, it will be added into Ω' as diverse routes with different costs. The solver IBM ILOG CPLEX is used to solve the LP-relaxed master problems and the SCP-based formulation of the master problem.

All instances have been solved using both heuristics 10 times on an Intel i7 PC with 8 cores à 3.4 GHz. The aggregated results for the IOTPP instances are given in Table 5.1. The cost values are given as the average cost of the best solutions found for the instances during the 10 trials. The time values given are the average time used in seconds to solve an instance in this set. For all instances, the best solutions found by the two heuristics are given in Table 5.2.

Table 5.1: *Results of IOTPP instances*

	ALNS		**HII-HET**		**HII-HOM**	
Ins.	cost	time	cost	time	cost	time
lc	2832.82	11.06	2791.88	8.90	2791.72	9.87
lr	4008.98	13.50	3966.38	10.27	3966.38	14.34
lrc	4294.03	15.64	4224.65	12.11	4224.65	14.21
all	3722.61	13.33	3673.13	10.35	3673.08	12.92

It is obvious that both combined approaches *HII-HET* and *HII-HOM* outperform the simple ALNS heuristic. For the tested IOTPP instances of the three sets lc, lr, and lrc with 100 customer nodes, the iterative approaches can achieve on average an improvement of 1.33% with regard to solution quality compared to the ALNS heuristic. *HII-HOM* performs slightly better than *HII-HET* since the latter failed to find the same best solution for instance lc104. Considering the computational time, the combined approach *HII-HET* appears to be favorable in comparison to the other two heuristics.

Table 5.2: Best found solutions of IOTPP instances

Ins.	cost	Ins.	cost	Ins.	cost
lc101	2689.23	lr101	4835.12	lrc101	4489.92
lc102	2800.18	lr102	4499.00	lrc102	4653.12
lc103	2815.91	lr103	4123.32	lrc103	3937.02
lc104	2697.19	lr104	3525.92	lrc104	3745.60
lc105	2905.68	lr105	4048.88	lrc105	4624.86
lc106	2785.80	lr106	4143.10	lrc106	4470.95
lc107	2796.16	lr107	3767.73	lrc107	3895.56
lc108	2784.02	lr108	3463.45	lrc108	3980.17
lc109	2851.33	lr109	4269.85	-	-
-	-	lr110	3639.20	-	-
-	-	lr111	3665.29	-	-
-	-	lr112	3615.75	-	-

5.4 Conclusions

In this chapter, two heuristic approaches are proposed to solve the complicated IOTPP introduced and modeled in the last chapter. The ALNS heuristic is an adjustment of the PDPTW heuristic proposed by Ropke and Pisinger (2006). The second one is more complicated. It follows the idea of column generation and iteratively calls the ALNS heuristic to construct new candidate routes. This heuristic differentiates itself from common approaches based on column generation in two aspects. First of all, instead of searching for a single route or several independently constructed vehicle routes at a time in the iterative column generation phase, a number of complete IOTPP solutions are generated while the total reduced costs of all routes in a solution are minimized. This strategy can also be understood in such a way that not only one vehicle route with the minimal reduced cost, but also those routes that are compatible with this route to construct whole IOTPP solutions are found at the same time. In other words, vehicle routes that have extremely low reduced costs but lead to a very bad partial solution for the other requests not in these routes as in the case of "cherry-picking" are not necessarily preferred in *HII*. The second aspect concerns the method used to get a feasible integer solution. Due to the high complementarity of the routes generated by solving the whole IOTPP instances during the route generation phase, many solutions to the LP-relaxed master problem of the tested instances are already feasible solutions to the original IOTPP. Even for the instances with fractional LP solutions, the gap between the LP objective function value and the solution obtained through solving the SCP-based problem is quite small (less than 2%), so that it is not necessary to call a B&B routine.

In order to evaluate the proposed heuristics, some theoretical instances are generated. Computational results show that the proposed heuristics perform almost equally well in solving the generated instances. The iterative approach performs slightly better than the

ALNS heuristic.

Later on in this thesis, these heuristics will be further used to solve larger instances in Chapter 8 in the research on IOTPP in CTP scenarios.

6 Collaborative transportation planning

Freight forwarders are confronted with increasing pressures to improve profitability, while it is difficult to further reduce operational costs. This problem is even more serious for those who already have almost exhausted their internal potential by processing optimization and applying new technologies including modern telecommunication equipment as well as powerful planning software. To further reduce cost and to increase operational efficiency, horizontal collaboration is proposed as a promising remedy for freight forwarding companies, especially for small and mid-sized ones. It is noticed that more and more horizontal cooperation initiatives are developing in practice (Cruijssen et al., 2007c) and this topic has attracted substantial interest of researchers in the last few years.

Early research on horizontal cooperation of independent freight forwarders exchanging requests can be found in Kopfer and Pankratz (1999), where such coalitions are referred to as *groupage systems*. In such coalitions, transportation planning is not executed by each participant separately but in a concerted fashion. This chapter is dedicated to the operational planning of independent freight forwarders in horizontal coalitions, which is referred to as CTP according to Wang and Kopfer (2014). Specifically, the combination of using both own vehicles and exchanging requests with partners in horizontal cooperations is considered.

CTP intends to improve the planning results of coalition members while preserving their autonomy. According to Stadtler (2009), collaborative planning can be understood here as a joint decision making process for aligning plans of individual coalition members with the aim of achieving coordination in light of information asymmetry. The specific goal of CTP is to achieve a reallocation of requests among the forwarders, with the effect that the total fulfillment costs are smaller than the sum of the forwarders' individual costs without collaboration. The obtained cost-savings present the joint benefits of the coalition that cannot be achieved individually. These joint benefits are then to be shared by the members in such a way that all participating freight forwarding companies in the coalition will improve their profitability.

This chapter is organized as follows. Section 6.1 offers an introduction on CTP including some illustrative examples explaining how cost reduction can be reached through collaborative planning. In order to realize the cost-saving potential embedded in the collaborative planning, appropriated request exchange mechanisms have to be developed. This problem

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_6,

is discussed in Section 6.2. Then, a mathematical model is presented to formulate the CTP in a more formal way in Section 6.3. Existing request exchange mechanisms in the CTP literature are reviewed in the last section.

6.1 Introduction

Cost reduction through CTP means to take advantage of both economies of scale and economies of scope. Economies of scale may be achieved by integrating several LTL requests in one tour, while economies of scope are reached by the combination of various tours which might decrease empty miles. The basic idea of CTP is to enable exchange of requests among the coalition members so that requests from different forwarders can be better bundled and thus, more efficient vehicle routes can be constructed for their fulfillment.

The effect of cost reduction through request exchange can be seen from the following examples. Figure 6.1 shows the situation, where transferring delivery requests between two freight forwarders may decrease the number of used vehicles. Without CTP, both forwarders A and B would have to utilize their vehicles to fulfill the customer requests. Suppose that the total demands of the four requests are less than one truckload. In this case, forwarder B can transfer his two requests to forwarder A for a certain price that is less than forwarder B's costs but higher than the additional costs for forwarder A. All four requests can then be fulfilled by using just one vehicle. However, it is worth mentioning that in this example, the goods to be delivered are homogeneous so that the customers can be served by either of the forwarders. This assumption makes it possible to use the VRP for the illustration.

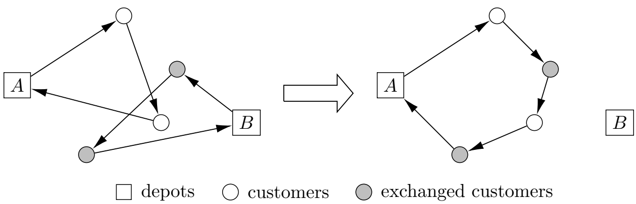


Figure 6.1: *Reduction of used vehicles, cf. Wang and Kopfer (2011)*

Figure 6.2 illustrates another scenario, where the routes of both forwarders A and B overlap. Through CTP they can exchange some requests and reduce the length of both routes. The total cost-savings achieved by performing CTP can then be shared between the two forwarders.

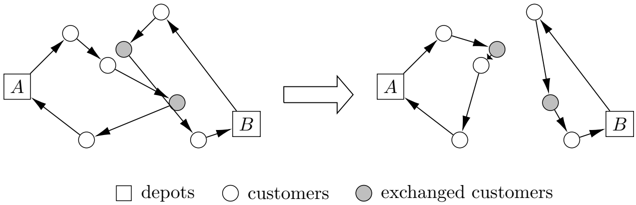


Figure 6.2: *Reduction of route length, cf. Wang and Kopfer (2011)*

A more complex possibility is to introduce transshipment points where goods associated with different transportation requests can be transferred between vehicles. Early research on transshipment in vehicle routing can be found in Mues and Pickl (2002) and Mues and Pickl (2005). This topic of synchronizing vehicles in vehicle routing has attracted more attention in recent years. Several new approaches are proposed in Petersen and Ropke (2011) and Masson et al. (2013) for the PDP with transfers as well as in Masson et al. (2012) for the DARP with transfers. Figure 6.3 gives an example of transferring customer requests between two collaborating freight forwarding companies at a transshipment point based on the VRPB. Forwarder A would deliver cargoes to D_1 and D_2 and then pick up goods at P_1 and P_2 before the vehicle goes back to the corresponding depot. One route of forwarder B with enough capacity lies nearby. B could pick up the loads at both customer nodes P_1 and P_2 in his route and transfer them at the transshipment point TP, which is the customer node P_3 in this example. In order to differentiate the routes of the two forwarders, the routes of forwarder B are depicted with dashed lines in Figure 6.3.

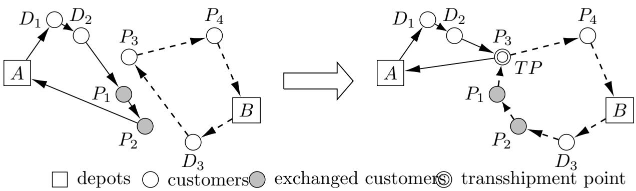


Figure 6.3: *Reduction of route length by introducing a transshipment point, cf. Wang and Kopfer (2011)*

Due to the cost-saving potential embedded in exchanging requests among forwarders, CTP has attracted substantial interest of researchers in the last few years. However, only discussions about the possibility of exchange complete requests as shown in Figures 6.1 and 6.2 can be found in the CTP literature. To the best of the author's knowledge, no research result considering transshipment in CTP has been presented yet.

Several studies are conducted to estimate the cost-saving potential embedded in CTP based on both theoretical and practical data sets. The commonly estimated cost reduction accounts to up to 30% (Cruijssen and Salomon, 2004; Cruijssen et al., 2007a; Krajewska et al., 2008). Cruijssen and Salomon (2004) and Krajewska et al. (2008) also report a decrement of total used vehicles between 7.3% and 10%. The standard method used in these studies to determine the cost-saving potential is to calculate the cost differences between the results of the *isolated planning* (IP), i.e., without any request exchange at all, and the *centralized planning* (CP), in which the routing problem is solved for the whole coalition as if it were one forwarding company. Some qualitative research on CTP has also been presented. Cruijssen et al. (2007b) conduct an empirical study to identify the potential benefits as well as some impediments of horizontal cooperation in logistics including performing CTP. Based on a case study and simulations, Cruijssen and Salomon (2004) discuss some influencing factors of request sharing and the impact of request sharing on clients, collaborating companies and the society. Wang and Kopfer (2011) analyze both the potential for cost reduction of freight forwarders in groupage systems and challenges for future research on CTP.

Although the benefit of CTP is widely recognized, forwarders still need to have efficient and proper approaches to realized the cost-saving potential embedded in CTP. Krajewska and Kopfer (2006) propose a general framework for the design of a complete CTP model which includes three phases: *preprocessing*, *profit optimization* and *profit sharing*. The main task of preprocessing is to identify customer requests suitable for exchange and to specify the payments for transferring them to partners. Profit optimization aims to find out a mapping between requests offered for exchange and collaborating partners to maximize the joint profits of the entire coalition. In the third phase, the joint profits achieved through exchanging requests are distributed among the partners according to a profit sharing scheme taking fairness criteria into account. The study conducted in this thesis focuses on the first two phases while profit sharing is considered as a separate challenging task, which is mainly to be tackled by developing adequate profit sharing strategies. A more comprehensive presentation of existing mechanisms of profit optimization proposed in CTP literature will be given in the last section of this chapter.

6.2 Design of request exchange mechanisms

In order to exploit the synergy effects embedded in CTP, appropriate request exchange mechanisms which are simple and implementable, yet effective in terms of generating high joint benefits have to be developed (Özener et al., 2011). Such mechanisms must be able to deal with distributed information and decision-making competences as well (Wang and Kopfer, 2014).

Synergy effect of performing CTP, i.e., the cost-savings can only be achieved when complementary requests of different participating forwarders are combined into request bundles. Thus, an important issue in the design of a request exchange mechanism is to deal with the construction of bundles. First of all, members of the coalition can theoretically take into account every possible combination of requests for exchange, which makes the number of bundles of requests to be considered exponentially large. Furthermore, it is very difficult for a forwarder to exactly evaluate his fulfillment costs for all possible combinations of requests independent upon other requests, especially for LTL pickup and delivery requests with time windows. This evaluation is the basis of determining the *ask price*, which is the lowest price the forwarder will charge for the execution of all requests included in a bundle. It seems that the ask price for a single request or a bundle of requests can be estimated by calculating the incremental cost, which is the difference between the total fulfillment costs of routing plans with and without this request/bundle. This approach presumes that everything else in the underlying tour remains unchanged. However, since the outcome of the exchange process is unpredictable, it is impossible for forwarders to know which requests will remain unchanged. In other words, it is not clear that based on which request portfolio the incremental cost of a request or a bundle should be calculated.

In the transportation service procurement problem (TSPP), a quite similar problem of bundling and evaluating requests is studied, for which a combinatorial auction (CA) is proposed instead of using a series of single-item auctions (Song and Regan, 2005; Lee et al., 2007). The bid construction problem in TSPP appears to be similar to the problem at hand, yet the ideas proposed in Song and Regan (2005) and Lee et al. (2007) are not applicable to the CTP scenario considered here. In the TSPP, shippers buy transportation services on several lanes from some carriers. A lane corresponds to a service on a transportation relation specified by an origin, a destination and a flow of goods which are to be transported from the origin to the destination during a predefined time interval. Different from an LTL request which has to be served only once on the operational level, a lane usually needs to be served frequently over a long period and thus belongs to the strategic planning level. Carriers can bid on these lanes for certain prices. The shippers choose those bids which minimize the total costs. Because bundling lanes may result in less empty miles as well as less travel and repositioning costs, shippers can reduce their procurement costs. Figure 6.4 illustrates this situation.

In this example, four lanes 1, 2, 3, and 4 have to be covered. On the one hand, since the first three lanes 1, 2, and 3 constitute a closed route without empty driven miles, carriers can bid on this bundle of lanes for a lower price than the total prices of three single-item bids, each containing only one of these three lanes. On the other hand, lane 4 can hardly

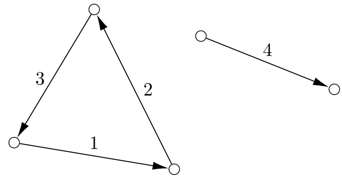


Figure 6.4: *Bundling of lanes in the TSPP (Wang and Kopfer, 2014)*

be combined with other lanes, it may be left unassigned in the CA and be auctioned later in a single-item auction.

In contrast to the TSPP, the problem of having requests unassigned may have serious consequences in the CTP scenario, since the coalition members are both seller and purchaser at the same time. Figure 6.5 illustrates this problem.

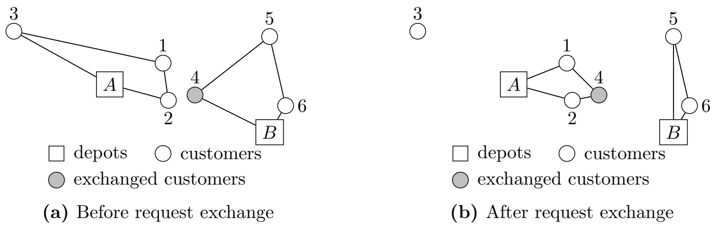


Figure 6.5: *Routing plans of collaborating forwarders, cf. Wang and Kopfer (2014)*

Suppose that two forwarders A and B try to exchange requests through a CA. Each of them has a vehicle that can serve up to 3 customers. Figure 6.5a shows that all six customers can be served before the exchange. Since request 4 lies near requests 2 and 3, forwarder A may bid on the bundle including requests 1, 2, and 4. Forwarder B may bid on the bundle of requests 5 and 6 since they can be well consolidated in his route. None of the two forwarder A and B would bid on request 3. The result would be that forwarder A wins the bundle with requests 1, 2, and 4, forwarder B wins the bundle with requests 5 and 6, and request 3 is returned to forwarder A. In this case, the total fulfillment costs for requests 1, 2, 4, 5, and 6 are reduced through the CA. However, because of the capacity limitation, forwarder A cannot fulfill in a single tour all four requests assigned to him (see Figure 6.5b). In order to get request 3 fulfilled, extra capacity is needed (e.g. by installing an additional tour or by subcontracting), which is usually very expensive and may shoot down the benefits reached by request exchange.

6.3 Mathematical formulation

In this section, the CTP problem studied in this and the following chapters is defined in a formal way. The scenario of a group of collaborating freight forwarders offering LTL pickup and delivery transport services is considered. It implies that the underlying routing problem is the PDPTW. Customer requests of all participating coalition members can be fulfilled by any vehicle of the coalition and thus can be exchanged among the forwarders. The planning situation considered here is static, which means that all information is available at the beginning of the planning. The dynamic CTP will be discussed later in Chapters 9 and 10.

Suppose that there exists a horizontal coalition of m independent freight forwarding companies. Each forwarder i comes along with a request portfolio R_i containing n_i, $i = 1, \ldots, m$ LTL pickup and delivery requests with time window restrictions, which are supposed to be offered for exchange in the coalition[1]. A fleet K_i with ϱ_i homogeneous vehicles in terms of both cost rates and loading capacity is positioned at the depot of each coalition partner i. However, the cost rates and loading capacities of vehicles need not be the same for different participants. All requests can be exchanged and fulfilled by any vehicle in the coalition.

The option of subcontracting is not considered in this chapter, so that it is assumed that in the IP scenario, all partners have enough capacity in their fleets; i.e., they can execute their whole original request portfolios R_i with their own fleets K_i. Participants are not expected to expose their private information to the agent nor to other partners.

The CTP problem together with the IP and the CP can be formally defined as follows. Each forwarder i can serve his requests R_i following his own routing plan Π_i with the costs of C_i by solving the PDPTW as defined in Section 2.2. For the sake of simplicity, the time dependent cost term can be omitted and the objective function for each single forwarder in the IP can be formulated as follows:

$$\min C_i = \sum_{k \in K_i} \alpha_k + \sum_{k \in K_i} \sum_{(u,v) \in A_i} \beta_k d_{uv} x_{uvk} \tag{6.1}$$

In this formulation (6.1), the arc set A_i is used instead of A_k for each vehicle $k \in K_i$ as the case in the PDPTW model presented in Section 2.2, which is defined based on the individual depots of the vehicles and the compatibility with the requests. In the CTP situation considered here, all vehicles of a single member are homogeneous and located at

[1]The phrases "forwarder", "member", and "partner" are used as synonyms in the context of CTP in this thesis.

the same start and end depots, so that the arc sets for all vehicles are the same and can be denoted simply as A_i.

The total execution costs of all forwarders in the IP are then given by $TC_{IP} = \sum_{i=1}^{m} C_i$. Consequently, TC_{IP} represents the upper bound for a CTP solution to be accepted. In the CP, a multi-depot PDPTW instance has to be solved for the entire request set $R = \cup_{i=1}^{m} R_i$. The total costs of the resulting routing plan Π for the entire coalition are denoted as TC_{CP}, which is the lower bound for the CTP solutions.

In the CTP scenario, the request portfolio R_i of forwarder i is completely offered for exchange. R_i can be divided into two parts. The first part is the set of requests $R_i^0 \subseteq R_i$ that have been offered but not transferred to other partners. The transferred requests constitute the set $R_i^- = R_i \setminus R_i^0$. The new request portfolio after the exchange is $R_i' = R_i^0 \cup R_i^+$, where R_i^+ is the set of requests that member i has acquired from other partners. The execution costs for the new request portfolio R_i' according to a plan Π_i' are given by C_i'. The CTP can be modeled as the following optimization problem:

$$\min\ TC_{CTP} = \sum_{i=1}^{m} C_i' \tag{6.2}$$

subject to:

$$R_i' \cap R_j' = \varnothing \quad \forall i, j = 1, \ldots, m, i \neq j \tag{6.3}$$

$$\cup_{i=1}^{m} R_i^- = \cup_{i=1}^{m} R_i^+ \tag{6.4}$$

The objective function (6.2) minimizes the total fulfillment costs of the entire coalitions. Restriction (6.3) ensures that each request is shifted exactly to one cooperation partner. Equation (6.4) ensures that all transferred requests are acquired by coalition partners.

6.4 Request exchange mechanisms in literature

Some approaches have been proposed in literature to tackle the challenging task of developing request exchange mechanisms for the CTP. These approaches differ not only in the solution methodologies but also in the CTP scenarios, especially in the underlying routing problems. In most approaches, auctions and similar mechanisms that enable exchange of bundles of requests are utilized.

Schönberger (2005) proposes a mechanism considering a variant of the PDPTW, where each partner has only one vehicle with unlimited capacity. Since hard time windows are considered, there are not enough capacities in the whole coalition for the execution of all requests. Requests that cannot be planned within the coalition have to be subcontracted to common carriers. An extremely high price level for forwarding the requests is assumed and it is tried to increase the degree of self-fulfillment. A CA is used as the exchange mechanism. Each participant solves a combined problem of request selection and PDPTW following his individual objectives. The resulting routes are proposed as bids and their costs as the bid prices. The winning bids are then chosen such that the total costs of subcontracting of the entire coalition are minimized.

Krajewska and Kopfer (2006) design a request exchange mechanism based on the concept of CA. They do not assume any specific routing problem for their CTP model. They presume that the fulfillment costs for any combination of requests can be exactly evaluated. However, the calculation of the potential fulfillment costs for all bundles of LTL pickup and delivery requests with time windows constitutes a very difficult problem which they do not consider. This problem increases even further when limitations of capacities are considered and post-auction planning is to be performed.

Schwind et al. (2009) propose two auction mechanisms for the request exchange among several warehouses of a single company. Since the goods to be delivered to the customers are of one single commodity, every customer can be served by any warehouse. Thus, the problem considered in this study can be modeled as the VRPTW. Due to the characteristics of the VRP, only requests located between neighboring profit centers are selected for exchange while the rest requests near the depots are kept by the warehouse and not offered for exchange. The proposed requests are then combined into bundles that can be fulfilled using one vehicle without violating the time window constraints. The resulted bundles represent the objects in the auctions. Each warehouse that is a member of the coalition solves its own VRPTW instances and calculates for each request bundle, in which he is interested, the marginal cost, which is declared as the corresponding bid price. The marginal cost is the cost difference between the routing results including and excluding the requests in the bid with those own requests that are not offered for exchange. Both a one-round auction and an iterative auction are tested and the results show that the iterative auction can achieve in general better results.

In the request exchange mechanisms developed by Berger and Bierwirth (2010) for inter-organizational scenarios, the exposure of information is limited. The authors consider the planning of pickup and delivery requests without capacity restriction, also known as the traveling salesman problem with precedence constraints (Renaud et al., 2000). Two auction mechanisms are proposed and tested. The first auction is a *second price sealed bid auction*, also known as the *Vickrey auction* (Vickrey, 1961), in which only one request is transferred at a time. For the second variant, which is a CA, each member in the coalition first identifies the most expensive request in his portfolio based on marginal cost evaluation and proposes it for exchange. Combinations of the proposed requests are then assigned through the winner determination of the CA. The whole process stops when no acceptable exchange can be identified. Computational results show that the CA outperforms the Vickrey auction, while the first one can realize an average cost-saving potential of 18.2-64.8% for different test sets.

A *route-based request exchange mechanism* is proposed by Wang and Kopfer (2014) for a coalition of freight forwarding companies serving LTL pickup and delivery requests. The authors also consider the practical capacity limitations of both single vehicle and the entire vehicle fleet. This approach is similar to the CA-based mechanisms as it also allows the exchange of bundles of requests. Specifically, only a part of all possible request bundles which are vehicle routes is considered. Computational results show that this approach can realize almost the complete cost-saving potential which is determined by solving the CP problem using effective heuristics. A more detailed presentation of this approach will be given in Chapter 7. This approach is further developed for an extension of the collaborative planning problem by considering also the different modes of subcontracting in Wang et al. (2014), which will be the topic of Chapter 8.

Besides the mechanisms that exchange request bundles, pair-wise exchange based approaches are also proposed in literature. An incentive compatible approach using cryptographic techniques for swapping pickup and delivery requests among independent carriers is proposed in Clifton et al. (2008). The authors develop a protocol that is secure against a centralized model referred to as the "trusted broker" model. This centralized model is actually the same as the CP defined previously in Section 6.3, since all parties have to give their input to the broker and the broker computes and returns the result.

Özener et al. (2011) study the lane exchange among FTL carriers and propose bilateral exchange mechanisms based on the calculation of the marginal costs of serving each single lane. Their computational experiments show that for the relevant setting to our scenario *no information sharing with side payments*, their approach can only realize about 30% of the potential cost-savings.

From a methodological point of view, the approaches proposed by Clifton et al. (2008), Schwind et al. (2009), Berger and Bierwirth (2010), and Özener et al. (2011) depend on the calculation of the marginal costs for each single request. The basic idea is to choose those requests with the highest marginal costs and offer them for exchange. If an exchange resulting in a better solution for all parties is found, it will be accepted and the exchange process continues. Otherwise the process ends. This idea suffers from the fact that the underlying process resembles a hill-climbing strategy, which does not accept any declined solution and thus cannot escape from local optima. The approaches in Schönberger (2005), Wang and Kopfer (2014), and Wang et al. (2014) follow the decomposition principle proposed by Dantzig and Wolfe (1960). In these approaches, the CTP problems are decomposed into several subproblems reflecting the routing decisions of single participants and a coordinating problem in form of a CA and an SPP/SCP. Using this decomposition scheme, each member in the coalition decides only for his part without regarding the feasibility of any other part (Dantzig and Wolfe, 1960) and without having to expose private information.

7 A route-based request exchange mechanism for the collaborative transportation planning

In this chapter, the *route-based request exchange mechanism* proposed by Wang and Kopfer (2014) to solve the collaborative routing problem of horizontal coalitions of independent freight forwarding companies introduced in the last chapter is described. In the scenario studied in this chapter, both the own capacity and those of the members in horizontal coalitions are considered. The main challenges of our design are to ensure that exchanged bundles can be exactly evaluated, to guarantee the protection of private information, and to deal with the difficulties caused by introducing capacity restrictions.

This request exchange mechanism aims to generate CTP solutions whose efficiency is close to that of CP, while the complexity of the request exchange process is relatively low and the possible negative influence on post-exchange planning illustrated in Figure 6.5b is already considered within the mechanism itself. This mechanism concentrates on complete routes instead of each request or arbitrary bundles, since the fulfillment costs of complete routes can be exactly calculated rather than for those later ones whose costs can only be estimated.

This chapter is organized as follows. A detailed description of the route-based request exchange mechanism of Wang and Kopfer (2014) is given in Section 7.1. Computational experiments are then reported in Section 7.2. Section 7.3 concludes this chapter.

7.1 The route-based request exchange mechanism

The route-based request exchange mechanism for the CTP follows the decomposition principles proposed by Dantzig and Wolfe (1960). The intention is that sensitive information like cost structure, customer payment and planning preferences is kept private by different participants. That is why it is essential to specify the information which actually should be communicated during the planning process in order to enable a successful decentralized planning. The entire routing problem of the coalition is decomposed into several routing problems and a coordination problem. Each routing problem is solved by a single partner for his own fleet while the coordination problem is formulated as an SPP/SCP and has to be solved by an agent of the coalition who practically can just be a computer. The

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_7,

planning process is designed as an interaction between participating forwarders and the agent. The forwarders generate vehicle routes independently and submit these routes to the agent of the coalition. Based on the routes submitted by the forwarders, the agent solves the coordinating problem of looking for a composition of winning vehicle routes that minimizes the total fulfillment costs. The forwarders iteratively generate and submit new routes based on the feedback information from the agent, which is deduced from the dual values of the LP-relaxed SPP.

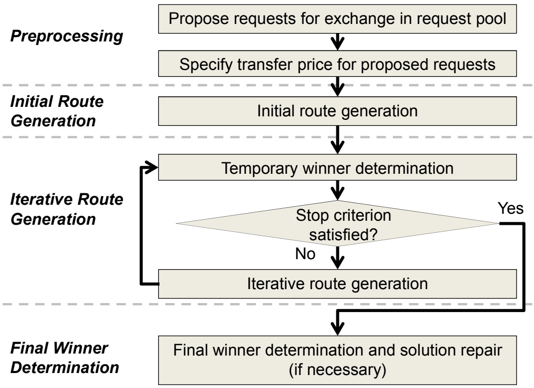


Figure 7.1: *Overview of the route-based request exchange mechanism (Wang and Kopfer, 2014)*

Figure 7.1 gives an overview of the entire process of the route-based request exchange mechanism. In the preprocessing phase, participating forwarders propose their requests that can be exchanged with other partners in a common pool. Moreover, a transfer price has to be specified and will be paid to the agent if someone else in the coalition intends to execute these requests. After that, all partners generate and submit some candidate vehicle routes to initiate the iterative route generation process. For each candidate route, the ask price has to be specified based on the fulfillment costs. In the iterative route generation phase, the problem of temporary winner determination modeled as an SPP is solved aiming to minimize the total fulfillment costs of all requests. The dual values related to the requests are then obtained by the agent while solving a linear relaxation of the SPP and these values are given to the forwarders, who can generate and submit new candidate vehicle routes iteratively until a certain stop criterion is satisfied. In the next step, the final winning routes are chosen by a final winner determination. For the execution of the winning routes, the ask prices will be paid to the forwarders by the agent. The difference between the total transfer prices paid to the agent and the total ask prices of winning routes paid by the agent will be determined as joint benefits of the coalition.

7.1.1 Preprocessing

In this thesis, the decision problem which is related to the selection of requests to be offered for exchange is not discussed. Instead, it is assumed that all partners $i = 1, \ldots, m$, offer their entire request portfolios R_i for exchange. Alternatively, it can be assumed that all partners have identified the request portfolios R_i which they want to offer for exchange in advance on the basis of a preliminary vehicle routing. In this situation, $R_i^* \supseteq R_i$ represents the entire set of requests acquired by forwarder i from his customers. In case of $R_i \subset R_i^*$, it is important to postulate that, with respect to the preliminary vehicle routing, none of the requests $r \in R_i$ is combined in a tour with any request $r' \in R_i^* \setminus R_i$. This means that within a preliminary planning each partner identifies very efficient routes which he wants to fulfill without request exchange and offers all other requests that are not contained in these routes for exchange.

As a consequence, the problem that collaborating partners have to solve in the preprocessing phase is to specify the transfer prices for their own request sets R_i. It seems to be desirable that the forwarders can specify the transfer price for each single request $r \in R_i$. However, to specify the exact cost as its transfer price for each single LTL request which is fulfilled together with many other ones in a common route is impossible. We thus design a mechanism which only needs an aggregated transfer price for the whole request portfolio R_i proposed by participant i. In order to determine the transfer price, each participant just needs to solve a PDPTW for the own requests he offers for exchange. The objective function of the PDPTW is given by:

$$\min C_i = \sum_{k \in K_i} \alpha_k + \sum_{k \in K_i} \sum_{(u,v) \in A_i} \beta_k d_{uv} x_{uvk} \tag{7.1}$$

For the fulfillment of his own initial request portfolio R_i within the coalition, participant i will not be willing to pay more than the transfer price C_i. The transfer prices are known but kept sealed by the agent. The sum of transfer prices of the request portfolios of all partners is the total fulfillment costs of the IP TC_{IP}. Based on the transfer prices of all partners, the coalition will accept only those CTP solutions with $TC_{CTP} \leq TC_{IP}$. As long as a CTP solution is accepted, the joint profits can be calculated as $TC_{IP} - TC_{CTP}$.

An important piece of information that must be transferred to the agent is the maximal number of routes that a participant can be assigned; i.e., the number of vehicles in the participant's own fleet available for the CTP. The agent can therefore make sure in the determination of winning routes that no partner will be assigned more routes than his fleet capacity allows.

After all partners have proposed their sets of requests, the request pool $R = \cup_{i=1}^{m} R_i$ is complete. All requests in the pool R are then visible to all members of the coalition and the route generation phase starts.

7.1.2 Initial route generation

During the route generation process, forwarders face two important questions: which requests in the pool R should be chosen for their own vehicles available for the CTP, and how these requests should be bundled into candidate vehicle routes. Theoretically, forwarders can consider all vehicle routes as long as they are feasible. For common problem sizes, this is not practical due to the very large number of possible routes. Moreover, the efficiency of a participant's plan Π_i' will be strongly diluted if some requests are covered by several of his winning routes but can be actually executed in only one route. Thus, instead of generating vehicle routes singly, entire PDPTW solutions are generated to take advantage of the complementarity of these routes the same as in the heuristic *HII* for solving the IOTPP in Section 5.2. By solving the PDPTW heuristically, a set of good but different solutions can be obtained at a time. The routes in the PDPTW solutions are then submitted for winner determination.

Since each freight forwarder can fulfill only a part of the requests in the pool R, he has to select the requests he wants to serve and create candidate vehicle routes for the fulfillment of the selected requests. The problem to be solved is thus a combined request selection and routing problem (see e.g. Butt and Ryan, 1999 or Feillet et al., 2005).

At the beginning of the route generation process, forwarders only know which requests are in the pool. Without knowing any payments for requests, they have to generate efficient routes with their costs as low as possible. The planning goals in this *initial route generation* step can thus be specified as firstly to get enough requests for their fleets, and secondly to generate efficient routes, which are similar to those of "cherry-picking". The first goal strives to increase the use of the own fleet, while the second one makes their candidate routes competitive and helps exhaust more cost-saving potential. Accordingly, for the combined request selection and routing problem in this step, the primary objective is to include as many requests as possible in the routes and the secondary objective is to reduce the routes' costs. This combined problem of request selection and the PDPTW is denoted as RSPDPTW1.

In order to define this problem, a "penalty cost" γ_r, $r \in R$ is introduced, which will be charged if request r is not planned in any route. A binary variable z_r^p, $r \in R$, representing whether a request is part of a route or not, is added to the model. If request r is not planned in any route, z_r^p will be one. Denote the new arc set for forwarder i as A_i' that is defined based on the entire request portfolio of the coalition R and his own vehicle fleet

K_i, the objective function of the RSPDPTW1 can be formulated in the following way:

$$\min \sum_{k \in K_i} \alpha_k + \sum_{k \in K_i} \sum_{(u,v) \in A'_i} \beta_k d_{uv} x_{uvk} + \sum_{r \in P} z_r^p \gamma_r \tag{7.2}$$

Requests with higher penalty costs will be preferred to be integrated into routes of the CTP compared to those having less penalty costs. An extremely high value will guarantee that the corresponding request will be planned in some route, as long as the capacity restriction holds.

After this problem has been solved in a heuristic manner, a set of good but different solutions are generated. All vehicle routes in these solutions are submitted for the winner determination with their costs as ask prices. For each route k submitted by member i, the ask price can be formally defined as:

$$p_k = \alpha_k + \sum_{(u,v) \in A'_i} \beta_k d_{uv} x_{uvk} \tag{7.3}$$

7.1.3 Temporary winner determination

When no forwarder wants to submit any further candidate routes, the agent temporarily solves the current winner determination problem (WDP) to provide useful information for the *iterative route generation* step (see Section 7.1.4). Suppose that n requests are offered for exchange, $R = \{1, \ldots, n\}$, and each forwarder i has submitted ω_i candidate routes. We add a fictive route for each single request in R with a very large ask price p_{max} to make sure that the WDP always has feasible solutions. The total number of candidate routes is $\omega = \sum_{i=1}^{m} \omega_i + n$. Let $a_{rj} = 1$ indicate that request $r \in R$ is held by route j and $a_{rj} = 0$ otherwise. The ask price for route j is p_j, $j = 1, \ldots, \omega$. The WDP can be modeled as an SPP by introducing a binary variable y_j, $j = 1, \ldots, b$, where $y_j = 1$ indicates that route j is chosen as a winning route:

$$\min \ TC_{CTP} = \sum_{j=1}^{\omega} y_j p_j \tag{7.4}$$

subject to:

$$\sum_{j=1}^{\omega} a_{rj} y_j = 1 \quad \forall r = 1, \ldots, n \tag{7.5}$$

$$y_j \in \{0, 1\} \quad \forall j = 1, \ldots, \omega \tag{7.6}$$

Since we want to consider the capacity restrictions of freight forwarders, this model has to be extended. Let $f_{ij} = 1$ indicate that a route j is submitted by freight forwarder i and $f_{ij} = 0$ otherwise. Participant i has ϱ_i vehicles in his fleet. We add the following constraint to the above model:

$$\sum_{j=1}^{\omega} f_{ij} y_j \leq \varrho_i \quad \forall i = 1, \ldots, m \tag{7.7}$$

The objective function (7.4) and restrictions (7.5)-(7.7) constitute the SPP based model of the WDP. Denote this model as WDP-SP. A linear relaxation of the WDP-SP is given by replacing (7.6) with:

$$y_j \geq 0 \quad \forall j = 1, \ldots, \omega \tag{7.8}$$

Denote the relaxed model (7.4), (7.5), (7.7), and (7.8) as WDP-LP. This relaxed problem is to be solved in this step. Then, the dual values of constraint (7.5) can be used for generating new candidate routes.

7.1.4 Iterative route generation

The route generation problem is to build new candidate routes with negative reduced costs $\bar{c_j}$, which can be calculated as follows for a variable y_j:

$$\bar{c_j} = p_j - \sum_{r=1}^{n} \pi_r a_{rj} - \sum_{i=1}^{m} \sigma_i f_{ij} \tag{7.9}$$

Variables $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ represent the dual variables corresponding to the constraints (7.5) and (7.7), respectively. The objective value of the WDP-LP can then be reduced by letting these new routes with negative reduced costs enter the basis until no route with $\bar{c_j} < 0$ can be found. The dual values π_r of the requests, $r = 1, \ldots, n$, are read by the agent and sent back to forwarders in a revised form π'_r, which can be determined by introducing a predefined minimal value $\pi_{min} \geq 0$ as:

$$\pi'_r = \begin{cases} \pi_{min} & if \quad \pi_r < \pi_{min} \\ \pi_r & if \quad \pi_r \geq \pi_{min} \end{cases} \quad \forall r \in R$$

Consider the meaning of the dual values π'_r, $r = 1, \ldots, n$, for the forwarders. If route j is generated by a specific forwarder i, we have $f_{ij} = 1$ and $f_{hj} = 0$, $h = 1, \ldots, m, h \neq i$.

Thus, (7.9) reduces for a particular forwarder to:

$$\bar{c_j} = p_j - \sum_{r=1}^{n} \pi'_r a_{rj} - \sigma_i \tag{7.10}$$

To find a route with $\bar{c_j} < 0$ for a particular forwarder i is equivalent to finding a route with $-\bar{c_j} = \sum_{r=1}^{n} \pi'_r a_{rj} + \sigma_i - p_j > 0$. Note that p_j is the ask price, which is the cost of a route. The revised dual values π'_r can be seen as a "fictive payment" for the fulfillment of a request r and the first term is then the total earning of this route. The second term σ_i can be regarded as the fixed costs for the vehicle, because σ_i is the dual variable for constraint (7.5) and it will always be non-positive as long as the dual problem of the WDP-LP is solvable. This interpretation helps to understand how this new route generation problem can be converted into a routing problem. As a result, the above task can then be interpreted as finding out a vehicle route with positive revenue.

Normally, this task is done by solving an optimization problem with the objective of maximizing $-\bar{c_j}$. If a feasible solution to this problem with non-positive objective value is found, a new route is generated. In this route-based request exchange mechanism, however, the idea to generate several new candidate routes by solving an RSPDPTW instance that has been used in the heuristic *HII* to solve the IOTPP in Section 5.2 is used again. This means that ϱ_i routes are generated at once by forwarder i. In order to emphasize the routes, index k is used instead of j. The objective function of this problem can be formulated as follows:

$$\max \sum_{k=1}^{\varrho_i} \sum_{r=1}^{n} \pi'_r a_{rk} + \varrho_i \sigma_i - \sum_{k=1}^{\varrho_i} p_k \tag{7.11}$$

Hence, the new route generation problem is to solve a request selection and routing problem in order to maximize the resulting revenue, which is calculated by subtracting the route costs from the "fictive payments" π'_r for the planned requests in the routes.

The meaning of introducing the revised dual value $\boldsymbol{\pi'}$ can now be seen in a more straightforward way. The minimum value π_{min} gives requests with small or even negative dual values the chance to be inserted into routes, of course only if the insertion causes very little costs. These two strategies help to find not only "good" routes that will improve the WDP-LP objective values, but also routes that are complementary to those good ones.

This route generation problem is denoted as RSPDPTW2. The objective function (7.11) can be formulated in a way that the RSPDPTW2 is modified to a revenue maximization problem. The first term of (7.11), $\sum_{k=1}^{\varrho_i} \sum_{r=1}^{n} \pi'_r a_{rk}$, calculates the total "fictive payments" for all requests planned in routes and can thus be substituted by $\sum_{r \in R} \pi'_r (1 - z^p_r)$.

The third term $\sum_{k=1}^{\varrho_i} p_k$ gives the total route costs of the ϱ_i vehicles, which can be replaced through $\sum_{k\in K_i} \alpha_k + \sum_{k\in K_i} \sum_{(u,v)\in A'_i} \beta_k d_{uv} x_{uvk}$. As a result, the objective function of the RSPDPTW2 is the same as the following one:

$$\max \sum_{r\in R} \pi'_r(1 - z_r^p) - \sum_{k\in K_i} \alpha_k - \sum_{k\in K_i} \sum_{(u,v)\in A'_i} \beta_k d_{uv} x_{uvk} + \varrho_i \sigma_i \tag{7.12}$$

Denote the fixed cost and the variable cost terms in (7.12), i.e., the two terms $\sum_{k\in K_i} \alpha_k$ and $\sum_{k\in K_i} \sum_{(u,v)\in A'_i} \beta_k d_{uv} x_{uvk}$, as FC_i and VC_i, respectively. This objective function (7.12) can be reformulated as a cost minimization function like in the RSPDPTW1 by setting $\gamma_r = \pi'_r$, $\forall r \in R$. We have

$$\begin{aligned}
& \max && \sum_{r\in R} \pi'_r(1 - z_r^p) - VC_i - FC_i + \varrho_i\sigma_i \\
& = && \sum_{r\in R} \gamma_r(1 - z_r^p) - VC_i - FC_i + \varrho_i\sigma_i \\
& = && \sum_{r\in R} \gamma_r - \sum_{r\in R} z_r^p \gamma_r - VC_i - FC_i + \varrho_i\sigma_i \\
\Leftrightarrow\ & \min && VC_i + FC_i - \varrho_i\sigma_i + \sum_{r\in R} z_r^p \gamma_r - \sum_{r\in R} \gamma_r \\
& = && VC_i + FC'_i + \sum_{r\in R} z_r^p \gamma_r
\end{aligned}$$

where $FC'_i = FC_i - \varrho_i\sigma_i - \sum_{r\in R} \gamma_r$. This is why the same heuristic can be used to solve both the two problems RSPDPTW1 and RSPDPTW2. This heuristic will be presented later in Section 7.2.2. The newly submitted routes are added to the existing candidate route set. The Steps *temporary winner determination* and *iterative route generation* are repeated until some stop criteria are satisfied. The route generation phase is then concluded. Please note that although this phase may have some iterations, the whole process has only one round. This means that the final winning routes will only be decided once in the following Step *final winner determination*.

7.1.5 Final winner determination

In this phase, the WDP is modeled as an SCP by replacing constraint (7.5) with:

$$\sum_{j=1}^{\omega} a_{rj} y_j \geq 1 \quad \forall r = 1, \ldots, n \tag{7.13}$$

This problem, defined by (7.4), (7.6), (7.7), and (7.13), is then denoted as WDP-SC.

Additionally, to make sure that the WDP-SC always has feasible solutions, a fictive route containing a single request $r \in R$ with the ask price p_{max} is added to the set of candidate routes for all requests by the agent.

The reason to choose the SCP-based formulation for the WDP instead of further using the WDP-LP is the same as that explained in Section 5.2.3, which is to minimize the efforts needed to get a feasible solution to the WDP-SP if the solutions of the WDP-LP and the WDP-SC are not feasible to the WDP-SP. A large number of iterations of route generation needed in a B&B process can put great demands on the computational efforts as well as on the communications between the forwarders and the agent, which is undesirable in the collaborative planning context. On the contrary, as the relaxation of the WDP-SC allows each request to be assigned more than one winning route, a WDP-SC solution can be infeasible to the WDP-SP. Such infeasible solutions can easily be repaired to feasible WDP-SP solutions through removing all multi-assigned requests from all but one route.

However, since the multi-assigned requests cannot be simply reinserted into vehicle routes of the forwarders by the agent, who has no idea about the concrete scheduling of the routes. Another heuristic is thus used here to repair an infeasible WDP-SP solution obtained by using the WDP-SC in this way. Hoping to minimize the damage of the synergy effects embedded in the routes, we give the multi-assigned requests to those partners who have won most of them. Let the set of multi-assigned requests of freight forwarder i be R_i^{mul}, $i \in \{1, \ldots, m\}$ and the set of all multi-assigned requests be R^{mul}. The set of freight forwarders that have won at least one multi-assigned request is $FC^{mul} = \{i | i = 1, \ldots, m, \wedge R_i^{mul} \neq \varnothing\}$. The outline of this simple heuristic is shown as follows.

Algorithm 5 WDP-SC solution repair heuristic

```
while |R^mul| > 0 do
    sort FC^mul so that for all j < k ⇒ |R_{FC^mul[j]}| ≥ |R_{FC^mul[k]}|
    assign request in R^mul_{FC^mul[1]} to FC^mul[1]
    R^mul := R^mul \ R^mul_{FC^mul[1]}
    R_i^mul := R_i^mul \ R^mul_{FC^mul[1]}, ∀i ∈ FC^mul
    remove i ∈ FC^mul with R_i^mul = ∅ from FC^mul
end while
```

Then, the agent asks each forwarder for the total ask price of the entire set of requests he has finally won after the exchange and repair procedure, which is the overall costs of the forwarder's new routing plan. Finally, the agent actualizes the result of the WDP.

Besides the requests that are successfully assigned in the winning routes, some requests may be left unassigned. Eventually, these requests are likely to be outsourced to external common carriers. In this case, the agent will ask the prices for outsourcing and perform the subcontracting if the CTP solution is accepted by the coalition. The total costs of the WDP-SP solution, including both the winning routes' ask prices and the possible costs for subcontracting, will be compared with the IP results. The WDP-SP solution will only be accepted if positive joint benefits can be realized.

7.2 Computational experiments

The performance of the route-based request exchange mechanism for the CTP described in the last section will now be evaluated by means of computational experiments. Since we have introduced a new CTP scenario with LTL requests with time windows, we need to generate test instances for this scenario. The method used to generate new CTP instances will be introduced first. Next, the route generator that solves both, the RSPDPTW1 and the RSPDPTW2, will be presented. CTP results obtained by applying our mechanism are presented and discussed at the end of this section.

7.2.1 Test instance generation

CTP test instances are generated by combining different PDPTW benchmark test instances generated by Li and Lim (2001), while each of them represents the request set of an individual participant of the coalition. Several instances with the same characteristics (C, R, or RC) and size (100-cases) are combined together into a single CTP test instance. Before a PDPTW instance is inserted into a CTP instance, coordinates of all nodes of this PDPTW instance have to be adjusted with the same amount $(\Delta X, \Delta Y)$ in order to correspond to the locations of forwarders in different regions. The number of vehicles in the own fleet is given as the number of used vehicles in the best-known solutions obtained from five heuristics: the heuristic by Li and Lim (2001), the heuristic by Bent and van Hentenryck (2003), the heuristic by Ropke and Pisinger (2006), and two commercial heuristics including the one developed by SINTEF and the other by TetraSoft A/S. Data sets of these PDPTW instances and detailed results can be found on a web page maintained by SINTEF[1].

For the sake of simplicity, all vehicles are assumed to have the same capacity given in the original data and the cost structure is fixed by setting the variable cost rate for each distance unit $\beta_k = 1$ for each forwarder i and each vehicle $k \in K_i$. We further assume that all vehicles have the same fixed costs, so that the fixed cost term can be ignored during

[1] `www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark`

Table 7.1: *Test instance generation information*

Instance	m	Generation information				
C101	2	lr103 (0,0) [9]	lc105 (31,7) [10]			
C102	2	lc106 (0,17) [10]	lc108 (23,0) [10]			
C103	3	lc102 (0,15) [10]	lc107 (44,34) [10]	lc109 (13,0) [9]		
C104	3	lc101 (0,0) [10]	lc104 (2,24) [9]	lc105 (28,7) [10]		
C105	4	lc101 (5,56) [10]	lc104 (0,23) [9]	lc107 (47,42) [10]	lc109 (27,0) [9]	
C106	4	lc102 (0,8) [10]	lc103 (35,8) [9]	lc105 (17,33) [10]	lc108 (19,0) [10]	
C107	5	lc101 (3,10) [10]	lc103 (20,29) [9]	lc105 (38,57) [10]	lc107 (0,42) [10]	lc108 (29,0) [10]
C108	5	lc101 (0,9) [10]	lc102 (7,48) [10]	lc103 (43,34) [9]	lc105 (10,0) [10]	lc108 (28,6) [10]
R101	2	lr103 (0,0) [13]	lr110 (20,3) [10]			
R102	2	lr106 (0,0) [12]	lr107 (8,34) [10]			
R103	3	lr102 (0,22) [17]	lr111 (36,29) [10]	lr112 (19,0) [9]		
R104	3	lr107 (3,0) [10]	lr108 (0,47) [9]	lr110 (16,31) [10]		
R105	4	lr106 (0,19) [12]	lr108 (23,19) [9]	lr109 (5,58) [11]	lr112 (12,0) [9]	
R106	4	lr101 (3,10) [19]	lr102 (0,50) [17]	lr105 (36,0) [14]	lr111 (26,30) [10]	
R107	5	lr102 (0,19) [17]	lr105 (14,0) [14]	lr107 (31,22) [10]	lr110 (18,35) [10]	lr112 (3,47) [9]
R108	5	lr101 (0,34) [19]	lr107 (33,50) [10]	lr108 (47,55) [9]	lr111 (7,9) [10]	lr112 (26,0) [9]
RC101	2	lrc103 (0,0) [11]	lrc107 (13,23) [11]			
RC102	2	lrc105 (0,0) [13]	lrc107 (1,47) [11]			
RC103	3	lrc102 (0,19) [12]	lrc103 (12,40) [11]	lrc104 (28,0) [10]		
RC104	3	lrc104 (0,8) [10]	lrc105 (17,31) [13]	lrc106 (31,0) [11]		
RC105	4	lrc101 (0,10) [14]	lrc103 (15,29) [11]	lrc107 (30,45) [11]	lrc108 (22,0) [10]	
RC106	4	lrc102 (0,20) [12]	lrc104 (45,20) [10]	lrc106 (12,57) [11]	lrc107 (39,0) [11]	
RC107	5	lrc101 (5,15) [14]	lrc104 (20,32) [10]	lrc106 (41,0) [11]	lrc107 (0,39) [11]	lrc108 (53,48) [10]
RC108	5	lrc101 (0,17)[14]	lrc102 (3,38) [12]	lrc103 (29,20) [11]	lrc105 (24,54) [13]	lrc107 (12,0) [11]

the planning. Thus, we can set the fixed cost α_k to zero and only focus on the variable costs of the routes. This offers the possibility to directly use the benchmark solutions to determine the costs for the IP scenario and to minimize the deviation from the optimal solutions caused by applying heuristics. The detailed information about the generation of the test instances can be found in Table 7.1. In total, 24 CTP instances, containing 2 to 5 forwarders each, have been generated using the 100-case PDPTW instances in the sets LC1, LR1, and LRC1. The sets LC2, LR2 and LRC2 are not used because the gap between the results of IP and CP is very small in these instances. The extreme long planning horizon of sets LC2, LR2 and LRC2 permitting many more customers to be serviced by the same vehicle (Solomon, 1987) makes it possible to get equivalent good solutions in both IP and CP scenarios.

7.2.2 Route generator

In order to generate routes and to solve the routing problems for the reference scenarios, an LNS which is similar to the ALNS presented in Section 5.1 is used. The only difference is that the adaptive framework is not used here. The purpose of this modification is to get more promising solutions and in turn more good candidate routes by intensifying the search in each iteration of the SA process. Different from the case of heterogeneous vehicle fleet for which more candidate routes of different types can be derived from a single route, only one candidate route can be generated in the homogeneous fleet case. This leads to a reduction of the number of total candidate routes that can be derived from the same number of recorded best found solutions. In order to reach a high efficiency of each LNS run in the iterative route generation step in finding efficient candidate routes without increasing the number of solutions recorded, the average quality of the recorded solutions has to be improved while a diversity of them has to be guaranteed. This has been done through conducting a more thorough search by applying all insertion operators to the current solution after a removal heuristic has been executed. While the best solution is chosen as candidate solution for the next iteration, all other solutions are also recorded for route generation. Since we do not apply the adaptive mechanism, we set the probabilities for executing the three removal heuristics: worst removal, random removal and Shaw removal to 0.2, 0.4, and 0.4, respectively.

Since this modification may make the heuristic more myopic in the search process, we have tested our LNS heuristic to verify the modification. We ran the LNS heuristic on each of the 100-cases and 200-cases instances ten times by giving γ_r large values. The heuristic has the same objectives as the majority of other heuristics for solving the PDPTW in literature: firstly to minimize the number of vehicles and secondly to minimize the total distances. All of our best solutions have the same numbers of vehicles as the best-known ones reported by SINTEF. With respect to the total distances, we have found the same best-known solutions for 53 of 56 instances of the 100-cases and for 29 of 60 instances of the 200-cases respectively. The average deviation between the best-known solutions and our best solutions of all test instances in the 100-cases is 0.08%. For the 200-cases it is 1.14%. It implies that our modification performs only slightly worse than the original ALNS. However, for the two test instances LR2_2_10 and LRC2_2_3, we have found better solutions than those published by SINTEF. We improved the results for LR2_2_10 with 3 vehicles from 3323.37 to 3316.39 and for LRC2_2_3 with 4 vehicles from 2938.28 to 2934.98. The results indicate that the modified heuristic can offer solutions to the PDPTW test instances of high quality that is comparable to the original ALNS proposed by Ropke and Pisinger (2006) while it increases the number of solutions considered for route generation. Our heuristic for the PDPTW constitutes the basis for the development of the CTP approach. Later on in this paper, the LNS heuristic is used for the generation of benchmarks by providing very good sub-optimal solutions for CP. These sub-optimal solutions are lower estimations for CTP and are used for the performance evaluation of our CTP approach.

During the route generation process, up to θ of the best solutions found in the search process are recorded. Routes in these solutions are submitted. For the RSPDPTW1 and the RSPDPTW2, the objective is only to minimize the costs or to maximize the revenues respectively, regardless of how many vehicles are used.

7.2.3 Computational results

As upper bounds for our tests, we use the best-known values from literature to calculate the total costs TC_{IP} in the IP scenario. To estimate the lower bounds TC_{CP} (CP), we solve the multi-depot PDPTW using the above mentioned LNS heuristic. For each instance, the multi-depot PDPTW is solved three times and the algorithm runs $\Psi = 15,000$ iterations each time. For the collaborative planning, forwarders repeatedly solve the RSPDPTW1 and the RSPDPTW2 using the same algorithm during the route generation process. Tuning experiments resulted in the following parameter setting, which offers a fair trade-off between time and quality. Each time, the algorithm runs only $\Psi = 5,000$ iterations and the vehicle routes in up to $\theta = 300$ of the best solutions found in these 5,000 iterations are submitted as candidate routes. In the *initial route generation* step, the penalty cost pen_r is set to 400. The minimal revised dual value for requests π_{min} is set to 10. For the WDP, the ask price for the fictive routes p_{max} is set to 400. We only execute one trial on each instance for the collaborative planning. The route generation process is stopped after 10 iterations or when the improvement of the objective value of the WDP-LP is less then β percent. For our tests, β is set to 0.1%, 0.2%, 0.5%, and 1.0% for instances with 2, 3, 4 and 5 freight forwarders, respectively.

The results are shown in Table 7.2. The number of freight forwarders m is given in the second column and the number of all requests n in the third column. The values of the absolute cost-saving potential $\Delta TC_1 = TC_{IP} - TC_{CP}$ and the relative cost-saving potential $\phi_1 = 100 \cdot \Delta TC_1/TC_{IP}$ (%) are shown in the sixth and seventh columns. Columns eight to ten give the results of the CTP. The columns for $\Delta TC_2 = TC_{IP} - TC_{CTP}$ and $\phi_2 = 100 \cdot \Delta TC_2/TC_{IP}$ (%) show the absolute and relative cost reduction compared to the IP. The efficiency parameter $\eta = 100 \cdot \Delta TC_2/\Delta TC_1$ (%) shows the realized percentage of the cost-saving potential and thus how efficient the request exchange mechanism is. Note that the values TC_{CP} for the CP are high quality sub-optimal solutions generated by using the LNS heuristic which has been introduced and evaluated in Section 7.2.2. They are not necessarily identical to the optimal solution of the CP. Values for $\#it_{RG}$ are the numbers of route generation iterations and the values for $\overline{\tau}_{RG}$ are the average time used by one forwarder to solve the RSPDPTW1 and the RSPDPTW2 once. The evaluation of the fulfillment costs of the routing plan for the entire request portfolio in the *final winner determination* step is not counted in $\#it_{RG}$. All route generation problems are solved on an Intel Core i7 PC (8 cores à 3.2 GHz). Finally, τ_{SC} is the time elapsed to solve the WDP-SC using IBM CPLEX 12.2 on an Intel Core i5 PC (4 cores à 3.33 GHz). Both PCs run Microsoft Windows operation systems. We do not report the time used to solve the WDP-LP, since it can be solved very quickly. The longest time is only 6.3 seconds for 5 forwarders with totally 265 requests, 60 vehicles and 17,954 routes.

7.2.4 Discussion of results

The results indicate that the route-based request exchange mechanism works very well for CTP. For 17 of the 24 instances, we have found solutions that are equal to or better than those obtained by using the LNS heuristic for the CP. However, it must be mentioned that the decentralized approach makes more demands on computational time than the LNS. Considering the solution quality, the decentralized planning through the route-based request exchange mechanism has found obviously better solutions than the LNS heuristic particularly for the large instances in sets R and RC. It seems that, especially for the instances with more participants, the CTP results can be further improved by reducing β and performing more iterations of route generation $\#it_{RG}$ and by increasing θ to submit more candidate routes in each iteration. The test settings show a stable performance with acceptable computational efforts needed for the CTP.

Table 7.2: *Results of planning*

Instance			Isolated planning	Centralized planning			Collaborative planning						
	m	n	TC_{IP}	TC_{CP}	ΔTC_1	$\phi_1(\%)$	TC_{CTP}	ΔTC_2	$\phi_2(\%)$	$\eta(\%)$	$\#it_{BG}$	$\overline{\tau}_{BG}$(min)	τ_{SC}(s)
C101	2	105	1864.29	1691.63	172.66	9.26	1699.08	165.21	8.86	95.68	7	0.80	0.64
C102	2	106	1655.38	1539.82	115.56	6.98	1539.82	115.56	6.98	100.00	6	0.75	0.02
C103	3	159	2658.48	2391.34	267.14	10.05	2414.99	234.49	9.16	91.15	8	1.56	3.40
C104	3	159	2517.89	2194.32	323.57	12.85	2198.10	319.79	12.70	98.83	9	1.86	0.69
C105	4	212	3518.49	3041.51	476.98	13.56	3027.86	490.63	13.94	102.86	7	3.11	11.14
C106	4	211	3519.67	2867.63	652.04	18.53	2882.58	637.09	18.10	97.71	7	3.04	6.21
C107	5	264	4348.61	3586.74	761.87	17.52	3579.69	768.92	17.68	100.93	7	3.46	173.74
C108	5	264	4348.61	3601.71	746.90	17.18	3596.64	751.97	17.29	100.68	5	3.48	0.95
avg.C	-	-	3053.93	2614.34	439.59	13.24	2616.75	435.46	13.09	98,48	7.00	2.26	24.60
R101	2	104	2452.03	2269.32	182.71	7.45	2263.21	188.82	7.70	103.35	6	0.68	0.36
R102	2	104	2363.93	2311.77	52.15	2.21	2311.98	51.95	2.20	99.60	5	0.63	0.53
R103	3	160	3600.24	3444.36	155.88	4.33	3394.24	206.00	5.72	132.15	7	1.22	5.29
R104	3	154	3239.63	3041.63	198.00	6.11	3014.98	224.65	6.93	113.46	7	1.35	13.67
R105	4	208	4434.32	3960.58	473.74	10.68	3906.01	528.31	11.91	111.52	6	2.29	140.54
R106	4	215	5624.38	5112.16†	512.22	9.11	5038.12†	586.26	10.42	114.45	6	2.71	5.43
R107	5	265	6139.11	5211.12†	927.99	15.12	5031.74‡	1107.37	18.04	119.42	5	3.30	429.75
R108	5	262	5843.75	5332.08	511.67	8.76	5186.87†	656.88	11.24	128.38	6	2.79	72.26
avg.R	-	-	4212.17	3835.38	376.80	7.97	3767.88	443.78	9.27	115.29	6.00	1.87	83.48
RC101	2	106	2488.89	2378.99	109.90	4.42	2386.78	102.11	4.10	92.91	6	0.75	0.48
RC102	2	107	2867.77	2690.49	177.28	6.18	2696.04	171.73	5.99	96.87	5	0.63	0.34
RC103	3	160	3945.21	3544.73	400.48	10.15	3511.99	433.22	10.98	108.18	6	1.32	3.06
RC104	3	161	4190.75	3693.64	497.11	11.86	3650.29	540.46	12.90	108.72	8	1.48	0.22
RC105	4	211	5345.12	4680.19†	664.93	12.44	4539.76	805.36	15.07	121.12	6	1.91	17.42
RC106	4	213	5341.35	4881.84	459.51	8.60	4777.83	563.52	10.55	122.63	6	1.86	86.22
RC107	5	265	6639.51	5800.29	839.22	12.64	5596.27	1043.24	15.71	124.31	5	2.56	657.59
RC108	5	266	7393.38	6237.40	1155.98	15.64	6100.68†	1292.70	17.48	111.83	6	2.88	25.05
avg.RC	-	-	4776.50	4238.45	538.05	10.24	4156.88	619.04	11.60	110.82	6.00	1.67	98.80
avg.All	-	-	4014.20	3562.72	451.48	10.48	3513.83	499.43	11.32	108.20	6.33	1.93	68.96

† Solutions with one vehicle less used compared with the isolated planning. ‡ Solutions with two vehicles less used compared with the isolated planning.

Table 7.3: *Efficiency and computational effort for different instance sizes*

m	$\overline{\phi}_1(\%)$	$\overline{\phi}_2(\%)$	$\overline{\eta}(\%)$	$\overline{\#it}_{RG}$	$\overline{\overline{\tau}}_{RG}(\text{min})$	$\overline{\tau}_{SC}(\text{s})$
2	6.08	5.97	98.07	5.83	0.71	0.40
3	9.23	9.73	108.75	7.50	1.47	4.39
4	11.65	13.33	111.72	6.33	2.49	44.49
5	14.47	16.10	114.26	5.67	3.08	226.56

A summarized comparison of efficiency and computational effort for different instance sizes is shown in Table 7.3. It is obvious that the cost-saving potential also rises with increasing number of participants in a coalition, but it becomes increasingly difficult for heuristic approaches to solve the problem centrally. Instead, CTP appears to be a preferable solution strategy. The success of our mechanism is based on the ability of the route generator to find more local optima by intensifying the search process (see Section 7.2.2) without necessarily having found the global optimum. CTP can provide satisfactory good solutions even when the number of iterations (5,000) running the heuristic for the RSPDPTW1 and the RSPDPTW2 is relatively small compared with the number of iterations running the heuristic in the CP (15,000). Meanwhile, the average time used for route generation $\overline{\overline{\tau}}_{RG}$ as well as the average time used for the WDP-SC $\overline{\tau}_{SC}$ also increases. However, it is surprising that the average number of route generation iterations $\overline{\#it}_{RG}$ does not significantly change, although the stopping criterion has been changed slightly.

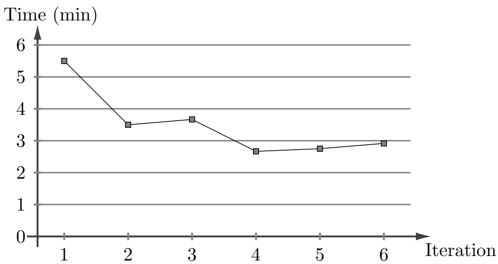


Figure 7.2: *Average time consumption for route generation of instance C108*

Another important observation is that τ_{RG} reduces significantly (up to 40%) in the late phase compared with the first iterations of the route generation process, because the route generator can be very well guided by the feedback information about the dual values. Requests with extremely low π'_r values will just be excluded from the consideration and forwarders can efficiently generate promising routes even while ignoring some requests in the pool. Figure 7.2 shows this trend using instance C108 as an example. If forwarders know which requests they do not need to consider in the route generation process, the time consumption used for route generation can be distinctly reduced. To achieve this, certain pre-selection rules can be developed and integrated into the mechanism. A possible rule is to exclude the requests located far away from the depots of the forwarders. Such pre-selection can be either performed by the agent who prepares for each forwarder a specific request portfolio, or by the forwarders themselves. It may further improve the performance of the proposed mechanism in terms of time consumption.

7.3 Conclusions

In order to increase the operational efficiency, small and mid-sized freight forwarders can build up horizontal coalitions and apply collaborative transportation planning techniques. Members of such coalitions can profit from the collaboration with partners without losing their autonomy. Through exchange of transportation requests, vehicle routes that are more efficient can be built and the total execution costs for the coalition can be reduced. In this chapter, the route-based request exchange mechanism is proposed to solve the routing problem in CTP of independent freight forwarders. This approach can deal with some difficulties that have been rarely considered at the same time in the previous research on CTP. Firstly, the underlying routing problem of the CTP scenario is the PDPTW. Secondly, the impacts of capacity restrictions on CTP have been dealt with explicitly. Finally, the degree of information exposure is low and decision-making competences within the coalition remain distributed.

Computational results show that this request exchange mechanism can implement the cost-saving potential embedded in the CTP to a great extent. For more than two-thirds of the instances, it even outperforms the LNS heuristic. Although the computational study seems to be a comparison of two heuristic approaches for the multi-depot PDPTW, the major difference between them is that the route-based request exchange mechanism is tailored for decentralized decision-making but the LNS heuristic is not. The comparisons shown in Table 7.2 are only meant to be considered as a measurement of the efficiency of our auction mechanism for the CTP, since clearly, using full transparency of the CP can in general lead to more powerful solution approaches than decentralized approaches which keep the autonomy of the planning partners.

8 Collaborative integrated operational transportation planning

Cooperation in transportation logistics offers freight forwarding companies new possibilities to increase the flexibility of their capacity and to reduce their operational costs. Such cooperation can happen between freight forwarders with subcontractors in vertical cooperation which is discussed in the IOTP in Chapters 4 and 5, or also between forwarders with equal partnership in a horizontal coalition as the case studied in Chapters 6 and 7. The consideration of the operational transportation planning of forwarders in these two types of cooperation makes it possible to study a more general case in this chapter, where all available resources, i.e., the own vehicle fleet, capacities of subcontractors including the common carriers, and the capacities of collaboration partners in horizontal coalitions through request exchange, are considered systematically in a holistic way in the operational transportation planning. Since this planning process is a combination of the IOTP and CTP, it can be referred to as the collaborative integrated operational transportation planning (CIOTP). It is endeavored in the CIOTP to use all available transportation resources efficiently and in turn to reduce the total fulfillment costs.

This chapter is organized as follows. Firstly, the planning problem of the CIOTP is defined in Section 8.1. Then, the solution approach proposed in Wang et al. (2014) which is an extension of the *route-based request exchange mechanism* described in Chapter 7 is described in Section 8.2. A computational study is then conducted in Section 8.3. The purpose of this study is to investigate how much the potential of cost reduction is and how efficient the extended route-based request exchange mechanism is in realizing this potential. Finally, some conclusions are drawn in Section 8.4.

8.1 Problem definition

Suppose that a horizontal coalition consists of m independent freight forwarding companies. The request portfolio of member i, $i = 1, \ldots, m$, is defined by R_i, which contains n_i LTL requests. The set of all requests of the coalition can then be defined as $R = \cup_{i=1}^{m} R_i$. Each request $r \in R$ can be subcontracted or exchanged in the horizontal coalition. In case that a request is decided to be subcontracted to a common carrier, a freight charge γ_r has to be paid, which is the same for all coalition members. It can be assumed that either all

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_8,

coalition members get the same quotation from the common carriers for the same request, or the lowest charge that the members can get from different common carriers is used for the whole coalition.

Considering different fulfillment modes in CTP means that each forwarder has a heterogeneous vehicle fleet consisting of vehicles of different modes, instead of the case studied in the CTP scenario in Chapter 7 where each forwarder has a homogeneous vehicle fleet. Let K_i^o, K_i^t, and K_i^d represent the sets of forwarder i's own vehicles, vehicles hired from subcontractors based on a tour basis, and vehicles hired on a daily basis, respectively. The corresponding fleet size is given by ϱ_i^o, ϱ_i^t, and ϱ_i^d, respectively. Then the entire vehicle set of forwarder i is defined by $K_i = K_i^o \cup K_i^t \cup K_i^d$. All these vehicles in K_i are stationed at the depot o_i of member i, which is the start and end node of their routes.

Similar to the discussion in Chapter 6, three relevant situations are to be specified. The first one is the IP case where each member forwarder solves his own IOTPP. Using the same notations introduced in Chapter 4.3, the objective function of the IOTPP for each forwarder in the horizontal coalition can be defined similar to (4.22) as follows:

$$\min C_i = \sum_{k \in K_i^o} \alpha_k + \sum_{k \in K_i^o \cup K_i^t} \sum_{(u,v) \in A_i} \beta_k d_{uv} x_{uvk} + \sum_{k \in K_i^d} \alpha_k z_k^d + \sum_{r \in R_i} \gamma_r z_r^c \tag{8.1}$$

The first term on the right-hand side of this objective function (8.1) calculates the total fixed costs of own vehicles. The variable costs of both own vehicles and the route costs of the vehicles paid on a tour basis are summed up in the next term. Flat-rates of vehicles hired on a daily basis will only be paid when these vehicles are used, i.e., $z_k^d = 1$. The last term represents the total costs for outsourcing the rest share of the request portfolio to common carriers. The binary decision variable z_r^c indicates whether a request is to be outsourced to a common carrier.

The total costs of the entire coalition in the IP scenario can be calculated by summing up all member forwarders' costs:

$$TC_{IP} = \sum_{i=1}^{m} C_i \tag{8.2}$$

The second relevant situation is the CP. The corresponding planning is done by solving a single multi-depot IOTPP instance for the entire coalition. In order to formulate the objective function of this problem, the sets of vehicles of different modes available in the entire coalition can be defined as $K^o = \cup_{i=1}^m K_i^o$, $K^t = \cup_{i=1}^m K_i^t$, and $K^d = \cup_{i=1}^m K_i^d$, respectively. Define the graph $G = (V, A)$ for this multi-depot IOTPP according to the requests and vehicles analog to the definition in Section 4.2, the objective function is given as follows:

$$\min TC_{CP} = \sum_{k \in K^o} \alpha_k + \sum_{k \in K^o \cup K^t} \sum_{(u,v) \in A} \beta_k d_{uv} x_{uvk} + \sum_{k \in K^d} \alpha_k z_k^d + \sum_{r \in R} \gamma_r z_r^c \tag{8.3}$$

The optimal results of these two planning scenarios represent the lower and upper bounds of acceptable CTP solutions, i.e., $TC_{CP} \leq TC_{CTP} \leq TC_{IP}$. In other words, the CTP solutions are only acceptable when they are better than the solutions obtained without horizontal cooperation. The small the gap between the solutions of the CP and the CTP is, the more efficient the applied CTP solution approach is.

The model (6.2)-(6.4) for the CTP presented in Section 6.3 can also be used here to define the CIOTP problem. The major difference is that the plans of the coalition members are made by solving the IOTPP instead of the PDPTW.

8.2 Solution approach for collaborative planning

In order to solve the CIOTP problem, Wang et al. (2014) extend the *route-based request exchange mechanism* presented in Chapter 7 for the heterogeneous fleet case. An overview of this mechanism is given in Figure 7.1.

The whole process begins with a *preprocessing* phase, in which participating forwarders propose their customer requests for exchange in a request pool. After that, all partners solve their individual IOTPP and the resulting costs are denoted as internal price C_i. The generated routes are reported to the agent in the form of request bundles. For each bundle, the route cost is revealed to the agent as the cost of this bundle but the concrete order of customer nodes in the corresponding route does not need to be reported. The agent solves the coordinating problem for the entire coalition by choosing the most promising bundles, i.e., by solving the SPP or the SCP, to minimize the total costs of the coalition. In order to improve the solution quality, the agent solves the LP-relaxed SPP and sends the dual values associated with the requests and vehicles back to the members so that they can generate new routes (bundles) using the feedback information. When a fairly well improved solution (compared to the IP scenario) is found, the chosen bundles are declared as winning bundles and their costs will be paid by the agent. Finally, the difference between the total internal prices paid to the agent and the total costs of winning bundles paid by the agent will be determined as joint benefits of the coalition.

8.2.1 Preprocessing

In this step, all members initially offer their request portfolios R_i, $i = 1, \ldots, m$, for exchange. As a result, $R = \cup_{i=1}^{m} R_i$ is the entire set of requests in the request pool. For each request $r \in R$, the information about the pickup and delivery locations, the time windows, and the load to be transported is also transferred to the agent and is generally accessible. However, the customer payments are held by the members who acquired them directly from their customers and do not need to be revealed to anyone else in the coalition, including the agent.

As it has to be guaranteed that the winning candidate routes chosen by the agent do not exceed the fleet capacities available to the forwarders, each partner has to report the maximum number of winning routes he is able to fulfill in each single mode, i.e., self-fulfillment, subcontracting on a tour basis, as well as subcontracting on a daily basis. Additionally, the modes are split into vehicle *types* because the fleet of a single mode could be heterogeneous as well in a more general sense. All vehicles of a single type are identical with respect to capacity and cost parameters. If a route is feasible for one vehicle of a specific type, then it is feasible for all vehicles of that type at the same cost.

After that, collaborating partners have to specify the internal prices for their own request sets R_i by solving their individual IOTPP. The sum of internal prices of the request portfolios of all partners corresponds to the total fulfillment costs TC_{IP} in the scenario IP . The internal prices are only necessary for the decision of the coalition on the acceptance of CTP solutions if $TC_{CTP} \leq TC_{IP}$, and for the determination of the collaborative profit $TC_{IP} - TC_{CTP}$. Apart from that they remain concealed.

8.2.2 Initial route generation

The first step of applying the decomposition scheme proposed by Dantzig and Wolfe (1960) is to generate a meaningful set of columns, i.e., vehicle routes in our case. This task is done in the Step *initial route generation* of the route-based request exchange mechanism. After that, the iterative process of route generation starts until some stopping criterion is met.

Purpose of this step is to generate a meaningful set of vehicle routes to start the iterative process of route generation. One strategy to achieve this is to encourage each partner to do “cherry-picking” for his own vehicles and all his available subcontractors, while leaving the remaining requests to common carriers. Thus, the objective function for initial route generation can be defined as maximizing the reduction of external freight charges for subcontracting requests to common carriers by doing “cherry-picking”. Denote the new arc set defined for forwarder i based on the entire request portfolio of the coalition R and his own fleet K_i as A'_i, the objective function of the routing problem to be solved by each coalition member can be formulated in the following way:

$$\max \sum_{r\in R} \gamma_r(1 - z_r^c) - (\sum_{k\in K_i^o} \alpha_k + \sum_{k\in K_i^o \cup K_i^t} \sum_{(u,v)\in A'_i} \beta_k d_{uv} x_{uvk} + \sum_{k\in K_i^d} \alpha_k z_k^d) \tag{8.4}$$

The first term calculates the amount of freight charges for the requests chosen for the fleet that will be paid to common carriers. The rest part calculates the route costs for fulfilling these request with available vehicles. Their difference is then the reduction of external freight charges through "cherry-picking". This is equivalent to:

$$\min \sum_{k\in K_i^o} \alpha_k + \sum_{k\in K_i^o \cup K_i^t} \sum_{(u,v)\in A'_i} \beta_k d_{uv} x_{uvk} + \sum_{k\in K_i^d} \alpha_k z_k^d + \sum_{r\in R} \gamma_r z_r^c - \sum_{r\in R} \gamma_r \tag{8.5}$$

This objective function (8.5) is actually the same as (8.1) because the last term in (8.5), i.e., $\sum_{r\in R} \gamma_r$ is a constant term and can be ignored. The only difference is that the request set R_i is substituted by the entire request set R of the whole coalition. Thus, this problem can be solved by using the ALNS heuristic presented in Section 5.1. The best solutions found during the search process will be recorded for the route generation.

It is worth mentioning that the operator used here to derive candidate routes from the recorded solutions works in the same way as the one used in the heuristic *HII* for the IOTPP (see Section 5.2). This operator also checks if a route in a solution is also feasible for other vehicle types besides its own assigned type in this solution. If it is true, then the same route will be submitted for the exchange as different candidate routes with different types, and in turn, different costs.

In the route-based request exchange mechanism described in Chapter 7, all vehicles of a single forwarder are homogeneous so that the fixed costs can be ignored in the consideration. In the scenario discussed in this chapter, however, the fixed costs of the own vehicles cannot be simply ignored any more. They are considered indirectly in the route costs. As a result, if a forwarder cannot win any bundles for some of his own vehicles, he must pay the fixed costs for these vehicles on his own. For the winning bundles, the costs will be compensated by the corresponding payments from the agent. This effect gives forwarders more incentive to generate efficient routes, especially for their own vehicles.

8.2.3 Temporary winner determination

After all members have submitted their bundles, the agent temporarily solves the current WDP that can be formulated in the following way. Suppose that in total $n = \sum_{i=1}^{m}$ requests have been offered for exchange and the set of these requests is $R = \{1, \ldots, n\}$. Each member i has submitted ω_i bundles. For each single request $r \in R$, a single-request bundle with its outsourcing price γ_r as its bundle cost is also added into the bundle set. The cost of a bundle c_j is identical to the route cost if it is derived from a vehicle route or to the outsourcing price if it is a single-request bundle. The total number of bundles is $\omega = \sum_{i=1}^{m} \omega_i + n$. Let $a_{rj} = 1$ indicate that request $r \in R$ is contained in bundle j, $j = 1, \ldots, \omega$, and $a_{rj} = 0$ otherwise. Denote the number of all vehicle types reported by all carriers as κ. Let $f_{gj} = 1$ represent that bundle j is proposed for type g, $g = 1, \ldots, \kappa$ and $f_{gj} = 0$ otherwise. The reported maximum acceptable number of winning bundles for type g is ϱ_g. The WDP can be modeled as an SPP while the binary variable y_j, $j = 1, \ldots, \omega$, takes value 1 if bundle j is chosen as a winning bundle:

$$\min \; TC_{CTP} = \sum_{j=1}^{\omega} c_j y_j \tag{8.6}$$

subject to

$$\sum_{j=1}^{\omega} a_{rj} y_j = 1 \quad \forall r \in R \tag{8.7}$$

$$\sum_{j=1}^{\omega} f_{gj} y_j \leq \varrho_g \quad \forall g = 1, \ldots, \kappa \tag{8.8}$$

$$y_j \in \{0, 1\} \quad \forall j = 1, \ldots, \omega \tag{8.9}$$

Denote this SPP-based WDP model as WDP-SP. An LP relaxation of the WDP-SP is given by relaxing y_j to be a continuous variable. Denote the relaxed problem as WDP-LP. The agent has to solve this relaxed problem in this step to get important information that can be used for generating new routes, especially the dual values associated with constraints (8.7) and (8.8).

8.2.4 Iterative route generation

The route generation problem is to build new routes with negative reduced costs $\bar{c}_j$, which can be calculated for a variable y_j as $\bar{c}_j = c_j - \sum_{r \in R} \pi_r a_{rj} - \sum_{g=1}^{\kappa} \sigma_g f_{gj}$, where $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ are the dual variables corresponding to the constraints (8.7) and (8.8), respectively. To improve the objective function value of the WDP-LP, new routes with $\bar{c}_j < 0$ have to be found. Values of $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ are read by the agent and sent back to forwarders while π_r is given in a revised form of $\pi'_r = \max\{0, \pi_r\}$, $r \in R$.

Given the values of $\boldsymbol{\pi'}$ and $\boldsymbol{\sigma}$, forwarders can generate new routes. In order to reduce the number of iterations needed for route generation and the associated communications between the agent and forwarders, a bunch of routes are generated at a time instead of searching for a route with the minimal reduced cost $\bar{c}_j$. Similar to the strategies presented in Section 7.1.4, this can be done by solving entire IOTPP instances. Since each member has a heterogeneous fleet, two options can be chosen analogues to those used in the heuristic *HII* for the IOTPP presented in Section 5.2. Each member can either solve an IOTPP instance for his entire heterogeneous fleet K_i (including vehicles of his subcontractors) or solve a set of IOTPP instances each only for a homogeneous vehicle set of a specific vehicle type. Denote these two options as *CTP-HET* and *CTP-HOM*, respectively.

Applying the *CTP-HET* option means searching for a set of vehicle routes, which can be executed by the entire fleet K_i of a single partner i. In order to emphasize that it is the vehicles that are to be found, index k instead of j in c_j can be used to formulate the objective function of this optimization problem:

$$\min \sum_{k \in K_i} \bar{c}_k = \sum_{k \in K_i} c_k - \sum_{k \in K_i} \sum_{r \in R} \pi'_r a_{rk} - \sum_{k \in K_i} \sum_{g=1}^{\kappa} \sigma_g f_{gk} \tag{8.10}$$

This objective function is equivalent to:

$$\min \sum_{k \in K_i^o} \alpha_k + \sum_{k \in K_i^o \cup K_i^t} \sum_{(u,v) \in A'_i} \beta_k d_{uv} x_{uvk} + \sum_{k \in K_i^d} \alpha_k z_k^d + \sum_{r \in R} \pi'_r z_r^c \tag{8.11}$$

Thus, the iterative route generation problem has exactly the same structure as the original IOTPP and we can use the same heuristic to solve it. During the search process, a number of solutions will be recorded for generating new bundles. For the option *CTP-HOM*, the objective functions for each vehicle type can be formulated directly in the same way as (8.11).

Again, the reason to use this strategy is to find not only "good" routes that will improve the WDP-LP objective values, but also routes that are complementary to those good ones. Computational results show that the gap between the WDP-LP and the final integer WDP-SP solutions is quite small (for almost all instances, the gap is smaller than 2%), so that it is unnecessary to call a B&B routine after the column generation phase. This effect is especially favorable for CTP, as such B&B routines would demand a lot of communications between the agent and forwarders and thus cause a lot of transactional costs.

The new routes are added to the existing bundle set. The Steps *temporary winner determination* and *iterative route generation* are repeated until some stopping criterion is met. The route generation phase is then concluded.

8.2.5 Final winner determination and flow of payments

In this phase, the agent solves another relaxation of the WDP-SP. Now, the constraint (8.7) is replaced by:

$$\sum_{j=1}^{\omega} a_{rj} y_j \geq 1 \quad \forall r \in R \tag{8.12}$$

This model, defined by (8.6), (8.12), (8.8), and (8.9), is the SCP-based model of the WDP. Denote this problem as WDP-SC.

If the WDP-SC solution is also feasible to the WDP-SP, the requests in each winning bundle will be assigned to the forwarder who has won this bundle. If a bundle representing the choice of outsourcing only one request to a common carrier wins, it will be returned to the member who proposed it for exchange.

If the WDP-SC solution is infeasible to the WDP-SP because some requests are assigned to several winning bundles, the agent can repair this solution simply by assigning such a multi-assigned request r to that partner who has won more multi-assigned requests than all other partners competing for r and removing it from all other bundles as described in Algorithm 5.

After the repair procedure, forwarders in the coalition are assigned the new request portfolios R'_i, $i = 1, \cdots, m$, including the (repaired) winning routes and the requests returned to him. These returned requests are the requests proposed for exchange but do not exist in any winning route. With the new request portfolio, corresponding internal payments to cover the fulfillment costs are also determined. The winning routes represent feasible routing plans for the members. However, as the winning routes are generated during the request exchange mechanism to minimize the coalition's overall costs, the resulted routing plans may not be the optimal solutions for the individual members' request portfolios R'_i. Forwarders can thus perform a post-exchange optimization by solving the IOTPP anew for themselves to get the costs for R'_i as C'_i as well as to improve their individual plans eventually. As a result, the ultimate fulfillment costs resulted by request exchange can be further reduced compared with the result of the WDP result. Finally, the agent inquires the total costs for the new request portfolio C'_i for all partners and gets the collaborative result as $TC_{CTP} = \sum_{i=1}^{m} C_i'$.

It is worth mentioning that associated with the payments, the responsibility of the fulfillment of the transferred requests is also shifted. A member who gets some requests from others must guarantee the fulfillment of these requests. However, they can choose either to fulfill them using their own vehicles or by subcontracting. Generally speaking, requests will only be transferred when they are included in some winning routes. But through the pose-exchange optimization in which both the requests in the winning routes and the returned requests are considered, it may be yet better to outsource some of the winning requests. In this case, the forwarders who have won these requests have to pay the freight charges to the common carriers on their own.

The total costs of the CTP solution, including both the winning bundles' costs and the possible costs for subcontracting, will be compared with the results of the IP. The CTP solution will only be accepted if positive joint benefits are realized.

If a collaborative solution has been accepted, the flow of payments among the coalition members can be determined. First, all forwarders have to pay the amount of the internal costs C_i to the agent. These are the costs that would result from the situation without collaboration. As an outcome of the exchange, requests will be assigned to forwarders according to the repaired winning bundles, and their costs will be paid to the members by the agent. Any request that has been assigned to a single-request winning bundle representing the option of common carriers will be returned to the partner who offered it for exchange. The freight charge for outsourcing a request r to common carriers is given by γ_r and also paid by the agent to the corresponding forwarder. The difference of the incoming payments acquired by the agent and the total payments paid out by him is then determined as joint benefits of the collaboration. They have to be shared among all participating members.

8.3 Computational Experiments

In order to analyze the cost-saving potential by performing CTP in the IOTP scenario and to evaluate the efficiency of the proposed CTP approach for the CIOTP, new theoretical test instances are generated. In the first step, the references for the CTP solutions, i.e., the results of the IP and the CP are obtained by using the heuristics for the IOTPP described in Chapter 5. Then, the extended request exchange mechanism is used to get CTP solutions.

8.3.1 Instance generation

Instances generated in Chapter 5 for the IOTPP (see Section 5.3.1) are used to generate CIOTP instances. They are constructed in a way similar to the generation of the CTP instances in Section 7.2.1 using the PDPTW instances. The collaborative instances are generated by combining the IOTP instances, while each of them represents the request set of an individual participant of the coalition. Three sets, i.e., C, R, and RC, are generated. Only instances sharing the same characteristics and size are combined together. For instance, all CTP instances of set C are generated using IOTP instance of set lc100. While inserting an IOTP instance into a CTP instance, the coordinates of the nodes in the IOTP instance are adjusted with the same amount $(\Delta X, \Delta Y)$, so that the IOTP instances in a single CTP are scattered representing the different locations of the participants. In total, 24 instances, containing 2 to 5 carriers each, are generated.

8.3.2 Isolated and centralized planning

In order to determine the cost-saving potential embedded in integrating CTP into IOTP and to get the reference planning results for the evaluation of the efficiency of the proposed CTP approach, the CIOTP instances are firstly solved for the IP and the CP scenarios.

For the IP, it is only necessary to sum up the fulfillment costs of each single instance used to generate the CIOTP instance that can be read from Table 5.2. The results are given in the third column in Table 8.3.

For the CP, the two approaches proposed in Chapter 5, i.e., the ALNS heuristic and the iterative heuristic with its two variations *HII-HET* and *HII-HOM* are applied. The parameter settings used previously in Section 5.3 are also used here. The aggregated results can be found in Table 8.1. The best found solutions for these CITOP instances are given in Table 8.2

Table 8.1: *Results of centralized planning for CTP instances*

	ALNS		HII-HET		HII-HOM	
Ins.	cost	time	cost	time	cost	time
C	9478.84	137.2	8765.16	115.9	8765.03	749.6
R	13710.64	133.7	12073.23	388.5	12073.39	3444.8
RC	14098.77	150.4	12702.17	141.1	12714.57	2293.6
all	12429.41	140.4	11180.18	215.1	11184.33	2162.7

Compared with the ALNS heuristic, the iterative approaches have achieved an improvement of solution quality for the CIOTP instances containing up to 500 customer nodes of 10.05% (*HII-HET*) and 10.02% (*HII-HOM*), respectively. Both variants iteratively update the evaluation of the requests and vehicles to navigate the ALNS heuristic to search

Table 8.2: *Best found solutions of CIOTP instances*

Ins.	cost	Ins.	cost	Ins.	cost
C101	5347.71	R101	7045.51	RC101	7414.69
C102	5339.95	R102	7504.16	RC102	8057.83
C103	7910.20	R103	10673.50	RC103	10798.98
C104	7315.82	R104	9824.52	RC104	11481.08
C105	9880.28	R105	13304.96	RC105	13716.46
C106	9709.52	R106	15480.89	RC106	14675.15
C107	12264.07	R107	15872.38	RC107	17200.33
C108	12305.16	R108	16829.24	RC108	18249.63

in different regions of the solution space. Through recombination of the partial solutions (routes) found during the process, better global solutions can be found.

For the three CIOTP instances R105, R107 and R108, it took quite a long time for CPLEX to solve the WDP-SC to optimality. However, if the *relative MIP gap tolerance* (MIP-Gap), which is the relative tolerance on the gap between the best integer objective and the objective of the best node remaining, is increased from 1E-6 to 0.01, this time can be reduced to a large extent. These three instances were then solved with this new setting again. The average computational time can be reduced up to 71.2% with an average decline of the solution quality of 0.2%. The computational time used by the combined approach *HII-HOM* increases dramatically with the number of customer nodes. During the search process, *HII-HOM* records much more routes than *HII-HET* using the same parameter (1,000 solutions each time). This leads not only to a longer computational time, but also to a considerable amount of data to be transferred to CPLEX, which becomes extremely time-consuming for larger instances with more than 3 forwarders.

8.3.3 Collaborative planning

In the first step *initial route generation*, all forwarders use in parallel the ALNS heuristic to generate IOTP solutions. In this step, the heuristic is allowed to run up to $\Phi = 10,000$ iterations and the computational time is restricted to 2 minutes. Each forwarder records up to 1,000 best solutions found during the search process and derives candidate routes from the routes in these solutions. After that, the step *iterative route generation* is repeated five times. The time limit to run ALNS is reduced to 30 seconds each time for *CTP-HET*. For *CTP-HOM* it is reduced to 20 seconds per vehicle type (1 minute for three types). In order to speed up *CTP-HOM*, only 350 solutions are recorded for route generation per vehicle type. Correspondingly, Φ is reduced to 2,500 iterations. No time limit is set to the CPLEX engine for solving the WDP-LP, since this can be done in less than 1 second for all instances. For the WDP-SC, however, the MIP-Gap is set to 0.01. A time limit of 2 minutes is also set for the CPLEX engine. As a result, the total time limits for *CTP-HET* and *CTP-HOM* are 6.5 and 9 minutes, respectively.

It has to be mentioned that the time limits are rather used for synchronizing the threads of computation of different coalition members. For the instances tested here, they are set so broad that they are not really used for stopping the ALNS heuristic. Due to the increased MIP-Gap for the WDP-SC, the time limit of 2 minutes for the CPLEX engine is rarely exhausted in the test.

When the new request portfolios R_i' for the coalition members $i = 1, \cdots, m$, have been specified, a post-exchange optimization starts. The heuristic *HII-HET* is then used in this step. The same parameter settings as reported in Section 5.3.2 are used here. The best results of instances obtained in the ten trials are then used to calculate the final result of the collaborative planning TC_{CTP}. For the most cases, a slight improvement can be achieved in this post-exchange optimization step so that the total fulfillment costs of the entire coalition are smaller than the results of the final winner determination.

The collaborative planning of the coalition for each CTP instance is simulated once. Thus, there are 24 samples for the evaluation of the performance of the proposed request exchange mechanism. The exact results can be found in Table 8.3. The total costs for the IP TC_{IP} are calculated by summing up the costs of all IOTP solutions. $\Delta TC_1 = TC_{IP} - TC_{CP}$ and $\phi_1 = 100 \cdot \Delta TC_1/TC_{IP}$ (%) are the absolute and relative cost-saving potential through the CP. Similarly, $\Delta TC_2 = TC_{IP} - TC_{CTP}$ and $\phi_2 = 100 \cdot \Delta TC_2/TC_{IP}$ (%) show the absolute and relative cost reduction through CTP. The efficiency parameter $\eta = 100 \cdot \Delta TC_2/\Delta TC_1$ (%) shows the realized percentage of the cost-saving potential and thus how efficient the request exchange mechanism is. Both variations of collaborative planning can realize the cost-saving potential to a great extend within the predefined time limit and perform equally well.

Comparing the different sets of instances, it is more difficult to realize the cost-savings using the proposed approach for the clustered instances (set C) than for the other two sets (sets R and RC). The geographical closeness of the customer nodes makes it more difficult for the decentralized decision-making process to find the optimal clusters for the forwarders.

8.4 Conclusions

The high volatility of market conditions has strongly influenced the operational transportation planning and changed the appearance of the resource management of modern freight forwarding companies. In order to face the new challenges and to improve their operational efficiency, forwarders have to reorganize their internal processes in order to

Table 8.3: *Result of collaborative planning*

Instance		**Isolated planning**	**Centralized planning**			**Collaborative planning**							
						CTP-HET				**CTP-HOM**			
	m	TC_{IP}	TC_{CP}	ΔTC_1	$\phi_1(\%)$	TC_{CTP}	ΔTC_2	$\phi_2(\%)$	$\eta(\%)$	TC_{CTP}	ΔTC_2	$\phi_2(\%)$	$\eta(\%)$
C101	2	5721.59	5347.71	373.88	6.53	5370.56	351.03	6.14	93.89	5396.73	324.86	5.68	86.89
C102	2	5569.82	5339.95	229.87	4.13	5339.95	229.87	4.13	100.00	5339.95	229.87	4.13	100.00
C103	3	8447.67	7910.20	537.47	6.36	8006.51	441.16	5.22	82.08	8062.91	384.76	4.55	71.59
C104	3	8292.10	7315.82	976.28	11.77	7376.73	915.37	11.04	93.76	7419.39	872.71	10.52	89.39
C105	4	11033.91	9880.28	1153.63	10.46	9976.86	1057.05	9.58	91.63	9991.40	1042.51	9.45	90.37
C106	4	10305.79	9709.52	596.27	5.79	10040.56	265.23	2.57	44.48	10066.69	239.10	2.32	40.10
C107	5	13991.00	12264.07	1726.93	12.34	12730.50	1260.50	9.01	72.99	12480.34	1510.66	10.80	87.48
C108	5	13995.02	12305.16	1689.86	12.07	12619.91	1375.11	9.83	81.37	12610.64	1384.38	9.89	81.92
avg.C	-	9669.61	8759.09	910.52	8.68	8932.70	736.92	7.19	82.53	8921.01	748.61	7.17	80.97
R101	2	7762.52	7045.51	717.01	9.24	7088.31	674.21	8.69	94.03	7071.23	691.29	8.91	96.41
R102	2	7910.83	7504.16	406.67	5.14	7504.16	406.67	5.14	100.00	7518.49	392.34	4.96	96.48
R103	3	11780.04	10673.50	1106.54	9.39	10697.86	1082.18	9.19	97.80	10773.10	1006.94	8.55	91.00
R104	3	10870.38	9824.52	1045.86	9.62	9854.24	1016.15	9.35	97.16	9840.90	1029.48	9.47	98.43
R105	4	15492.15	13304.96	2187.19	14.12	13592.17	1899.98	12.26	86.87	13746.51	1745.64	11.27	79.81
R106	4	17048.29	15480.89	1567.40	9.19	15501.19	1547.10	9.07	98.70	15570.66	1477.63	8.67	94.27
R107	5	19570.56	15872.38	3698.18	18.90	16292.19	3278.37	16.75	88.65	16639.18	2931.38	14.98	79.27
R108	5	19347.34	16829.24	2518.10	13.02	17000.83	2346.51	12.13	93.19	17187.08	2160.26	11.17	85.79
avg.R	-	13722.76	12066.90	1655.87	11.08	12191.37	1531.40	10.32	94.55	12293.39	1429.37	9.75	90.18
RC101	2	7832.58	7414.69	417.89	5.34	7421.39	411.19	5.25	98.40	7419.96	412.62	5.27	98.74
RC102	2	8520.42	8057.83	462.59	5.43	8105.12	415.30	4.87	89.78	8057.82	462.60	5.43	100.00
RC103	3	12335.74	10798.98	1536.76	12.46	10836.12	1499.62	12.16	97.58	10798.98	1536.76	12.46	100.00
RC104	3	12841.41	11481.08	1360.33	10.59	11491.63	1349.78	10.51	99.22	11497.36	1354.05	10.54	99.54
RC105	4	16302.67	13716.46	2586.21	15.86	13805.91	2496.76	15.32	96.54	13889.88	2412.79	14.80	93.29
RC106	4	16765.23	14675.15	2080.08	12.47	14981.50	1783.76	10.64	85.34	14908.08	1857.16	11.08	88.86
RC107	5	20582.20	17200.33	3381.87	16.43	17289.80	3292.40	16.00	97.35	17332.34	3249.86	15.79	96.10
RC108	5	21600.48	18249.63	3350.85	15.51	18556.74	3043.74	14.09	90.83	18682.52	2917.96	13.51	89.08
avg.RC	-	14597.59	12699.27	1898.32	11.76	12811.03	1786.57	11.10	94.38	12822.12	1775.48	11.11	95.45
avg.All	-	12663.32	11175.08	1488.24	10.51	11311.70	1351.63	9.54	90.49	11345.51	1317.82	9.34	88.87

better manage external relations to partners in both vertical and horizontal cooperation. In this chapter, the operational transportation planning of forwarders in road haulage is discussed taking own resources and those of partners into account. The two related topics, i.e., IOTP and CTP, are integrated into the planning process of forwarders. A planning framework that can help forwarders achieve higher operational efficiency by exploiting vertical and horizontal cooperation has been proposed. Computational results based on theoretical instances show that the systematical utilization of both internal and external resources for the execution of customer requests can result in favorable benefits. It has also been demonstrated that the proposed approach is highly efficient in realizing the cost-saving potential, while private information of coalition members in horizontal cooperation remains concealed.

9 Dynamic collaborative transportation planning

Through using efficient collaborative planning approaches and exchanging requests with other coalition members, freight forwarding companies can reduce their operational costs to a great extent as shown in the previous chapters in the static planning scenarios. In this chapter, using CTP for cost reduction in a dynamic environment is investigated. Compared to the static CTP problems, the performance of using CTP for dynamic planning has rarely been studied in literature yet. In order to fill this gap in research on CTP, the dynamic collaborative FTL PDP with forwarding (DCFPDPF) is introduced. In the dynamic scenario, new requests are released during the execution of vehicle routes. Since all information revealed for the planning is certain, the DCFPDPF is a dynamic and deterministic problem.

In order to solve the DCFPDPF considered in the following, two rolling horizon planning solution approaches that solve the static problems periodically based on the updated current information are proposed. The first one is introduced in Wang and Kopfer (2013a) and solves a new static problem when a predefined interval is reached. In the second approach which is proposed in Wang and Kopfer (2013b), the static CTP is triggered whenever a new request becomes urgent and must be irrevocably planned at that time. In both approaches, the *route-based request exchange mechanism* for the static CTP discussed in Chapter 7 is adapted and embedded into the rolling horizon planning frameworks.

This chapter is organized as follows. First, literature on dynamic and deterministic vehicle routing problems and on the dynamic CTP is briefly reviewed in Section 9.1. Then, the DCFPDPF is defined formally in Section 9.2. The two rolling horizon planning solution approaches for this DCFPDPF are presented in Section 9.3. Section 9.4 concludes this chapter.

9.1 Literature review

The dynamic CTPP extends the static CTP over a long time horizon with gradually revealed request information. The two closely related topics of this problem are the dynamic and deterministic vehicle routing and the CTP. In this section, the literature review focuses on some major contributions to dynamic and deterministic routing problems and to the dynamic CTP. Reviews on the static CTP can be found in Chapter 6.

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_9,

In contrast to static routing problems where all input data of the problems are known a priori at the time when the routes are constructed, some input data are revealed or updated during the period of time in which operations take place in a dynamic routing problem (Berbeglia et al., 2010). Moreover, a routing problem can be either deterministic or stochastic according to the information quality which reflects possible uncertainty on the available data (Pillac et al., 2013). The reader is referred to Pillac et al. (2013) for a recent review on dynamic routing problems. Specifically, Berbeglia et al. (2010) review the literature on dynamic pickup and delivery problems (DPDP).

Pillac et al. (2013) classify the solution approaches for dynamic routing problems into two categories. The first one comprehends *periodic re-optimization* approaches. After having constructed a set of initial routes, these approaches periodically solve static problems corresponding to the current states triggered by specific incidences. The first type of such triggers is an update of input data, which practically can be the release of a new customer request. This strategy is used in Psaraftis (1980), Yang et al. (1998), and Yang et al. (2004). Psaraftis (1980) proposes a dynamic programming approach for the dynamic DARP. Rolling horizon approaches are used to solve the real-time FTL PDPTW in Yang et al. (1998) and its extension by the possibility of rejecting requests and soft time windows in Yang et al. (2004).

Alternatively, a re-optimization can be trigged whenever a predefined interval is reached. Savelsbergh and Sol (1998) apply a branch-and-price algorithm to solve the static PDPTW for each single re-optimization. Similarly, Chen and Xu (2006) use the column generation scheme in a dynamic approach and solve the VRPTW each time. An ant colony system is developed in Montemanni et al. (2005) for a dynamic VRP in which encrypted information about characteristics of good solutions are conserved in a pheromone matrix and passed on to the next static problem after a predefined interval.

The second category of solution approaches is referred to as *continuous re-optimization*. The general idea is to run the optimization routines and maintain information on good solutions in an adaptive memory. Whenever an event occurs, e.g., a new request is known or a vehicle has finished its current job, a decision procedure stops the optimization routine and updates the routes. After that, the optimization routine is restarted with the new generated solutions. Different from periodic re-optimization approaches, drivers do not know the next customer to serve until they have finished their current job. Diverse optimization routines are used in these approaches. Gendreau et al. (1999) apply a parallel TS proposed in Taillard et al. (1997). Another TS is proposed in Gendreau et al. (2006) while the concept of ejection chains (Glover, 1996) is used to construct a powerful neighborhood structure.

In addition to work on developing efficient solution approaches for dynamic routing problems, the influence of different waiting strategies on the quality of solutions to dynamic routing problems is studied in Mitrović-Minić and Laporte (2004). Tjokroamidjojo et al. (2006) analyze how valuable it is for carriers to have the load information in advance.

As discussed previously in Chapter 6, some approaches are proposed to tackle the challenging task of solving the request exchange problem in the static CTP. On the contrary, little research was conducted on the dynamic CTP. Song and Regan (2003) study a static CTP problem of a coalition of freight forwarders fulfilling FTL pickup and delivery requests. Whenever a member acquires a customer request, he launches an auction for the assignment of this request and acts as an auctioneer. Other coalition members acting as bidders calculate the marginal costs of inserting this request into their existing routes. The request will be transferred to the bidder with the lowest bid price if this price is lower than the auctioneer's own marginal costs. Wang and Kopfer (2013a) propose a rolling horizon approach for the dynamic CTP with predefined time interval between two successive planning processes. The route-based request exchange mechanism described in Chapter 7 is adapted to solve the request exchange problem in each static planning. Their computational study shows that CTP is especially preferable in a highly dynamic environment. The authors further recommend to use advance information about requests and to plan in a forward-looking way for a better solution quality.

9.2 Problem definition

The DCFPDPF deals with a horizontal coalition of m independent freight forwarding companies, who offer FTL transportation services for their customers. Each forwarder in the coalition i, $i = 1, \dots, m$, has an own homogeneous fleet K_i with ϱ_i vehicles. At the beginning time $t_0 = 0$ of the entire time horizon $[0, \infty)$, these vehicles are located at different locations. No specific end depots are assigned for them in the dynamic situation. Let $K = \cup_{i=1}^{m} K_i$ denote the entire fleet of the coalition. Without loss of generality, it can be assumed that all vehicles in K are equally equipped so that every request can be fulfilled by any vehicle in K. However, all forwarders may use their own scheme to calculate route costs for their own fleet, i.e., the vehicles may have different variable cost rates β_k.

Each forwarder i in the coalition acquires requests from his customers during the entire time horizon. The set of requests of forwarder i over the entire time horizon is denoted as R_i. Let $R = \cup_{i=1}^{m} R_i$ denote the set of all requests to be served. A request $r \in R$ must be transported from its pickup location to the corresponding delivery location. At each location u, the operation (pickup or delivery) must be started in a customer defined time window $[a_u, b_u]$. The service time at u is given by s_u. In a static PDP, all request information is available at the time of planning. In a dynamic PDP, however, requests may be released while the routes are executed. The time when a request r is released to a forwarder is called the *release time* and denoted as t_r^{rls}.

In a dynamic environment it is important to define the corresponding request set associated with a point in time. At any point in time t, only a subset of requests $R_i^t \subset R_i$ is known to forwarder i. R_i^t is defined as the set of all requests acquired by forwarder i with their release time no later than t, i.e., $R_i^t = \{r \in R_i | t_r^{rls} \leq t\}$. Furthermore, the set R_i^t can be divided into two parts. The first part consists of all requests that have not been planned yet and is denoted as $R_i^{t,a}$, where the index a means that these requests are still "active" for the planning. The complement $R_i^t \setminus R_i^{t,a}$ is the set of requests that have already been irrevocably planned and thus are no more relevant to the planning process at t. These requests cannot be reassigned either because their services have already been started or even finished or because their services must be started soon after t.

In order to fulfill a customer request, a forwarder has three options. The first one is the *self-fulfillment*, i.e., to assign this request to one of the forwarder's own vehicle. The second one is to *subcontract* it to other freight carriers. The third one is to transfer it to other coalition members through request exchange. For the sake of simplicity, only the fulfillment mode *subcontracting to common carriers* introduced in Chapter 4 is considered and the phrase "subcontracting" is used as synonym for "subcontracting to common carriers" in this chapter. In case of subcontracting, a certain price γ_r must be paid. The DCFPDPF aims at minimizing the overall costs of the entire coalition to fulfill all requests in R.

9.3 Solution approaches

In this section, two rolling horizon planning approaches are proposed for the DCFPDPF. First, the two basic rolling horizon planning frameworks are presented. Then, some important issues related to the DCFPDPF are described in more detail. At last, the extension of the route-based request exchange mechanism that is embedded in the rolling horizon planning frameworks is introduced.

9.3.1 Rolling horizon planning with fixed interval

The approach used in Wang and Kopfer (2013a) is based on a rolling horizon planning framework with fixed interval, which is denoted as *RHP-INT*. Figure 9.1 shows the principle of this framework illustratively.

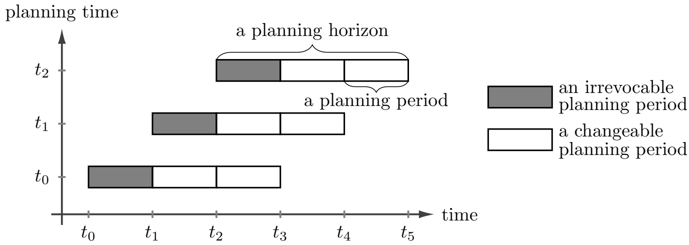


Figure 9.1: *Framework of rolling horizon planning with fixed interval (Wang and Kopfer, 2013a)*

The entire time horizon is divided into a series of *planning periods*. All planning periods have the same length τ. We use identifier p to denote the planning periods, $p = 0, 1, 2, \ldots, \infty$. Let H denote the number of planning periods contained in each *planning horizon*. The length of a planning horizon is then given by $L_H = H\tau$. At $t_0 = 0$, an initial plan Π_1 for the first planning horizon including planning periods $p = 1, 2, \ldots, H$ is constructed. The plan for the first planning period is irrevocable. The plan for the next planning periods $p = 2, \ldots, H$, however, will be actualized in the forthcoming plans as a result of the dynamically released new requests during the execution of planned routes. After that, at the end of each planning period p, i.e., at planning time $t_p = p\tau$, $p = 1, 2, \ldots, \infty$, a new plan Π_{p+1} for the next planning horizon ranging from planning periods $p+1$ to $p+H$ will be made based on the updated status. Again, the partial plan constructed for planning period $p+1$ will be irrevocable while the rest part will be kept changeable. Figure 9.1 shows the case when H is set to 3.

At each planning time t_p, $p = 0, 1, 2, \ldots, \infty$, new customer requests can be assigned to each vehicle which has finished all the requests that have been planned irrevocably in its route in the previous planning periods. In case that a vehicle has already finished serving all assigned requests, it waits at the location of the last customer until new orders are assigned to it.

9.3.2 Request triggered rolling horizon planning

Another possibility of doing rolling horizon planning is to adjust the plan after each actualization of the status of the request portfolio. This variant is referred to as *RHP-RT*. Figure 9.2 illustrates the framework of this option.

At $t_0 = 0$, an initial plan Π_1 for the first planning horizon with a predefined length L_H

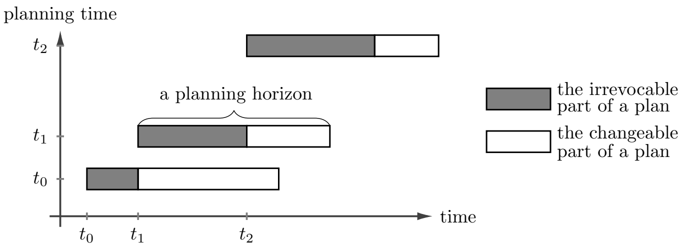


Figure 9.2: *Framework of request triggered rolling horizon planning (Wang and Kopfer, 2013b)*

is constructed. At t_1, the status of the requests is changed and a new plan is made. The part of the first plan Π_1 from t_0 to t_1 has already been executed and thus is the irrevocable part of Π_1. The remaining part, i.e., from t_1 to $t_0 + L_H$ will be actualized due to the new plan. At t_2, the status of the request portfolio changes again, and the process repeats. As a result, the irrevocable part of a plan in *RHP-RT* is dynamically determined through the actualization of the status of the request portfolio.

Different triggers can be used within this framework. In Song and Regan (2003), the trigger is defined as the release of a new request. In their approach, so long as a new request is known to the coalition, the member who acquires it initiates a single-item auction, through which this new request will be irrevocably assigned. Another option that will be used here is to initiate an exchange process when a request becomes urgent and must be irrevocably planned immediately. This means that the process is not related to the release time of the requests, but to the *due time* of the requests. The due time t_r^{due} of a request r is defined as the time that is t^{ld} time units earlier than the latest time when the service of this request must begin, i.e., the end of the time window of the related pickup operation e_{r^+}. Given the predefined *lead time* t^{ld} for the planning, the due time of a request r can be calculated as $t_r^{due} = e_{r^+} - t^{ld}$. Figure 9.3 gives an illustrative explanation of the definition of the due time of requests, where the two rectangles represent the time windows of the pickup and delivery operations of request r.

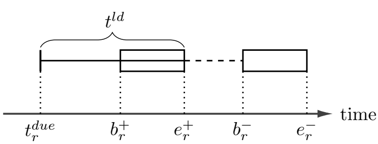


Figure 9.3: *Due time of a request (Wang and Kopfer, 2013b)*

9.3.3 Determination of due requests

In a dynamic environment, customer requests are released as time goes on. Some requests may be released shortly before the latest time when the service must begin and thus requires a quick response of the forwarders. Others may be known quite a long time before the time window opens and may leave the forwarders much more time and possibilities to seek for the best plans. Thus, it is necessary in a rolling horizon planning approach to differentiate requests according to their urgency.

Let $R_i^{t,a}$ denote the set of all active requests of forwarder i at the planning time t. An active request at a specific point in time is a released yet not executed request that has not been irrevocably planned previously. The requests in $R_i^{t,a}$ can be further differentiated into two types according to their urgency. The most urgent requests are labeled as *due requests* that must be irrevocably planned at t, and the remaining part of requests are called *non-due requests* that can be planned at t, but do not need to be irrevocably planned. Let $R_i^{t,d}$ and $R_i^{t,n}$ denote the sets of due requests and non-due requests, respectively, and we have $R_i^{t,a} = R_i^{t,d} \cup R^{t,n}$. Then, the sets of all active, due, and non-due requests of the entire coalition at t can be defined as $R^{t,a} = \sum_{i=1}^{m} R_i^{t,a}$, $R^{t,d} = \sum_{i=1}^{m} R_i^{t,d}$, and $R^{t,n} = \sum_{i=1}^{m} R_i^{t,n}$, respectively.

Due requests are differently defined in the two different rolling horizon planning frameworks. In case of *RHP-RT*, a new planning process will be launched when one or more requests become due requests. Here, a request r is defined as a due request at time t if $t_r^{due} \leq t$ holds. More precisely, a request r triggers a new planning when $t_r^{due} = t$. The requests that trigger a new planning process are the due requests in this plan.

In the *RHP-INT*, due requests are determined in another way. Obviously, at the planning time $t = p\tau$ which is the beginning time of the $(p+1)^{th}$ time period , all requests that must be picked up in the $(p+1)^{th}$ planning period are due requests. I.e., the set of due requests $R^{t,d}$ at t consists of the requests whose service must be started before the end of the $(p+1)^{th}$ planning period at their pickup locations. Additionally, in order to improve the continuity of the plan, requests whose pickups must be served soon after the end of the $(p+1)^{th}$ planning period are also considered as due requests in *RHP-INT*.

9.3.4 Planning strategies using advanced request information

In a dynamic environment, forwarders can improve the quality of their planning if they are offered the request information in advance. However, the value of the advanced request information is not always the same according to how much in advance the information is released (Tjokroamidjojo et al., 2006). For a given request, the shorter the time is from the current point in time to the latest time allowed to begin the service, i.e., the more urgent the request is, the more valuable the information about this request is. On the contrary, the longer the time is and the less urgent the request is, the less valuable the related request information is for the planning at the current time. Thus, it is important in a rolling horizon planning to specify planning strategies that differentiate the importance of the known requests to the current planning according to their urgency.

Based on the differentiation of requests according to their due time, planning strategies using advanced request information can be defined based on two factors. The first one is the length of the planning horizon L_H. The longer this parameter is, the more forward-looking the planning is. Specifically for *RHP-INT*, the minimum value of L_H equals to the length of a planning period τ and the planning focuses only on the most urgent requests. Thus, for the planning with fixed interval, any strategy with $L_h = \tau$ is referred to as *myopic planning* (MYP), and any strategy with $L_H = H\tau$, $H > 1$, is denoted as *forward-looking planning* (FLP). For *RHP-RT*, MYP means that only one request is considered when a static plan is made, i.e., only the costs of moving a vehicle from its current position to the pickup location of this request and bringing the goods then to the corresponding delivery location are considered. In this case, L_H is no more a predefined constant but decided dynamically by the due time of the requests.

The second factor is a weight function that assigns each considered request a weight for the planning that reflects its urgency. This weight can also be interpreted as an evaluation of the advanced information of requests associated with a specific point in time. The weight function has a simple form in any MYP: since only due requests are considered and all due requests are the same important at the planning time, they shall have the same weight value. In FLP, the situation is more complicated since the plan for the non-due requests will be actualized in one or more successive planning periods. Measuring the urgency of a request for planning can be done by multiplying the freight charge for outsourcing with different a weight, which is dependent on the time when the request must be served. It is worth mentioning that by setting the weight of requests that are to be served after the time $t + L_H$ to zero, the term $R_i^{t,a}$ can still be used for the rolling horizon planning framework with a constant L_H. Thus, we also use $R_i^{t,a}$ to denote all known requests in the next planning horizon at any planning time t. In the collaborative planning scenario, these two strategies can be formulated by using the corresponding notations $R^{t,d}$ and $R^{t,a}$.

9.3.5 Identification of requests for exchange

A planning strategy using advanced request information specifies the requests to be considered in each static planning and the evaluation of their urgency. For CTP, however, it must be determined additionally which requests are the candidates for exchange.

For *RHP-RT*, all requests that have triggered the new planning at the same time are candidates for exchange, since a CTP will be launched when some requests become due and must be irrevocably planned at that moment.

In case of *RHP-INT*, two situations have to be differentiated: MYP and FLP. As MYP only deals with requests in $R^{t,d}$, all requests in $R^{t,d}$ are candidates for exchange. In other words, they are considered in the planning and have to be irrevocably assigned to the forwarders through request exchange. For FLP, however, all known requests $R^{t,a}$, i.e., all active requests which currently known for the next H planning periods are included for constructing the plan. Nonetheless, only the most urgent requests, i.e., the requests in $R^{t,d}$ that must be irrevocably planned, will be candidates for exchange. Actually, MYP can be realized by setting the weight for all requests in $R^{t,n}$ to zero and thus be regarded as a special case of FLP in a broader sense.

It is important to differentiate the requests to be considered and those to be exchanged in case of FLP. The reason is that each reallocation of requests is associated with transfer payments among the members which are determined based on the costs of routes. These costs are supposed to be as accurate as possible. In a dynamic environment, these costs can only be precisely determined for the partial routes serving the most urgent requests since they will not change during their execution. Another important reason is that although request information in advance should be considered in planning, the plan for requests that are not urgent should not be fixed as soon as the plan is made (Tjokroamidjojo et al., 2006).

9.3.6 Extended route-based request exchange mechanism

In order to solve the DCFPDPF, the route-based request exchange mechanism that is described in Chapter 7 is extended to solve the static problem periodically within the rolling horizon frameworks.

Each time a new planning is initiated at time t, all partners first propose all their active requests $R_i^{t,a}$ into the common request pool of the coalition in the *preprocessing* stage. The request set of the coalition $R^{t,a} = \cup_{i=1}^{m} R_i^{t,a}$ is then divided into $R^{t,d}$ and $R^{t,n}$. The set $R^{t,d}$ is also the set of requests to be assigned by the static CTP.

After the requests have been proposed, each forwarder i solves a routing problem considering only his own fleet and set of requests $R_i^{t,a}$. Through introducing a weight w_r^t for each request $r \in R_i^{t,a}$, the objective function of this routing problem can be formulated as follows:

$$\min \sum_{k \in K_i^t} \sum_{(u,v) \in A_i^t} \beta_k d_{uv} x_{uvk} + \sum_{r \in R_i^{t,a}} \gamma_r w_r^t z_r \tag{9.1}$$

Let $K_i^t \subseteq K_i$ be the set of vehicles that are available in the current planning. A_i^t is the set of arcs defined by forwarder i's own requests and vehicles and d_{uv} is the distance from u to v, $(u,v) \in A_i^t$. The decision variable $x_{uvk} \in \{0,1\}$ indicates if an arc (u,v) is used in vehicle k's route and the other binary variable z_r indicates if a request is outsourced to a common carrier. In case of MYP, all requests in $R_i^{t,n}$ have $w_r^t = 0$ and all due requests have $w_r^t = 1$. In case of FLP, all due requests also have a weight of one while other requests have a weight less than one. Next, each forwarder declares the total costs for his own request portfolio $R_i^{t,d}$ as a transfer price, which is the maximum payment for the fulfillment of these requests without cooperation. In case of FLP, a route may have both due and non-due requests. Only the first part containing the due requests is considered, i.e., the partial route costs until the delivery location of the last due request in a route are used. Denote this transfer price as $C_i^{t,d}$. The total transfer prices $TC_{IP}^{t,d} = \sum_{i=1}^{m} C_i^{t,d}$ represent the total costs for due requests at t without cooperation. This information is used for the acceptance of CTP solutions, which will only be accepted when they are better than the solutions of IP, i.e., $TC_{CTP}^{t,d} < TC_{IP}^{t,d}$.

The next step is the *initial route generation*. Each forwarder i solves a routing problem for his own available vehicles K_i^t and generates a set of routes fulfilling the requests selected from the request pool $R^{t,a}$. The objective function is the same as (9.1) except that the sets A_i^t and $R_i^{t,a}$ are replaced by A^t and $R^{t,a}$, respectively:

$$\sum_{k \in K_i^t} \sum_{(u,v) \in A^t} \beta_k d_{uv} x_{uvk} + \sum_{r \in R^{t,a}} \gamma_r w_r^t z_r \tag{9.2}$$

Through solving this problem in a heuristic manner, a set of different solutions can be obtained. The first part of the routes in these solutions containing only the due requests is reported as candidate routes to the agent. The costs of the partial routes will be declared as the costs of these candidate routes.

After the set of candidate routes has been initialized, the iterative process starts. In each iteration, the agent solves the WDP in the form of an LP-relaxed SPP to minimize the total fulfillment costs of all due requests. That the requests can be outsourced to common carriers is also considered in the WDP in such a way that each due request

$r \in R^{t,d}$ is either to be assigned to a winning candidate route or common carriers for the price γ_r. This price is the same for all coalition members.

Suppose that there are n^t requests in $R^{t,d}$ and b_i candidate routes have been proposed by forwarder i in the *initial route generation* step. For each request r, a fictive route representing the common carrier option with the route cost of γ_r is also added into the set of candidate routes. Thus, there are $b = \sum_{i=1}^{m} b_i + n^t$ candidate routes in total. Let $a_{rj} = 1$ indicate that request r is in route j and $a_{rj} = 0$ otherwise, $j = 1, 2, \ldots, b$. We use $f_{kj} = 1$ to indicate that route j is proposed for vehicle k, $k \in K^t$. The cost of a candidate route is denoted by c_j. The binary decision variable y_j, $j = 1, 2, \ldots, b$ indicates whether a route is chosen as a winning route. The WDP can be formulated as follows:

$$\min TC^t_{CTP} = \sum_{j=1}^{b} c_j y_j \tag{9.3}$$

subject to:

$$\sum_{j=1}^{b} a_{rj} y_j = 1 \quad \forall r \in R^{t,d} \tag{9.4}$$

$$\sum_{j=1}^{b} f_{kj} y_j \leq 1 \quad \forall k \in K^t \tag{9.5}$$

Constraints (9.4) ensure that each request is assigned to exactly one winning route and constraints (9.5) ensure that each vehicle is assigned to at most one route. The agent solves the linear relaxation of this model and gets the dual values related to (9.4) for requests π_r and related to (9.5) for vehicles σ_k. These values are sent back to the forwarders for the next iteration of route generation.

Using this feedback information, forwarders can generate and submit new candidate routes in the *iterative route generation* step by solving another routing problem with the following objective function:

$$\min \sum_{k \in K_i^t} \sum_{(u,v) \in A^p} \beta_k d_{uv} x_{uvk} + \sum_{r \in R^{t,d}} \pi_r z_r + \sum_{r \in R^{t,a}} \gamma_r w_r^t z_r \tag{9.6}$$

Again, only the first part of each route in the solutions obtained by using heuristic algorithms is proposed as a candidate route. The ALNS heuristic presented in Section 5.1 is used to generate candidate routes in both the initial and iterative route generation steps. Iterative route generation stops when predefined criteria are satisfied.

The final step of the whole process is the *final winner determination*. The agent solves an SCP-based formulation of the WDP by replacing (9.4) with:

$$\sum_{j=1}^{b} a_{rj} y_j \geq 1 \quad \forall r \in R^{t,d} \tag{9.7}$$

If some requests belong to more than one winning route in the WDP solution, the agent calls a simple repair routine to obtain a feasible solution to the CTP problem. The result of the WDP will only be accepted if its total costs are less than the total transfer prices reported by all members in the first phase, which indicates a cost reduction for the entire coalition.

9.4 Conclusions

In order to improve their operational efficiency, small and mid-sized freight forwarders are suggested to integrate external transportation resources into their operational planning process. Besides the option to outsource requests to common carriers, they can further reduce their costs through setting up horizontal coalition with fellow companies and perform CTP. The static CTP has been studied for different situations in the last decade. However, little research has been conducted to study CTP in a dynamic environment.

Compared with the static CTP, the consideration of the dynamic CTP is a more challenging task in the academic research on transportation logistics. In order to fill the gap in the research on dynamic CTP, the DCFPDPF is introduced and formally defined in this chapter. Although the static problem of the DCFPDPF is the FTL PDPTW, which is relatively simple compared with the PDPTW for LTL requests, the study of the DCFPDPF can help better understand the CTP of forwarder coalitions in dynamic environments and aims to appeal to more intensive studies in this research area.

In order to solve the DCFPDPF, two rolling horizon approaches are proposed in this chapter, too. The first one is proposed in Wang and Kopfer (2013a). The second one is proposed in Wang and Kopfer (2013b). The difference between these two approaches is the trigger of new planning processes. In *RHP-INT*, a new planning is triggered by the fixed interval defined as the time span between two consecutive planning processes. In *RHP-RT*, a new planning process is triggered by the actualization of the status of requests. Both strategies have been used in literature to solve dynamic routing problems. In the next chapter, a comprehensive computational study on the DCFPDPF is conducted to evaluate the proposed approaches and to derive some practical suggestions for forwarders.

10 Computational study on the dynamic collaborative transportation planning

In this chapter, a comprehensive computational study including several tests is conducted to obtain some insights into the dynamic CTP. Specifically, the tests are designed to answer the following three questions: Firstly, how much cost-saving potential can be realized by using the approaches proposed in Chapter 9. Secondly, how the overall planning results are influenced by different factors such as planning strategies and characteristics of the instances. Last but not least, how the realized cost-savings through CTP against IP are affected by these factors.

The first test described in Section 10.1 is reported in Wang and Kopfer (2013a). This test aims at recognizing the relationship between the degree of dynamism of the instances and the performance of the *RHP-INT*. The results of this test are analyzed in more detail in this section. Then, the four tests described in Wang and Kopfer (2013b) are presented in Sections 10.2 to 10.5. These tests are designed to analyze the influence of some other factors on the planning results. Some conclusions of this computational study are drawn in Section 10.6.

10.1 Test 1: Dynamism of instances

An important character of the dynamic vehicle routing is the degree of dynamism of the instances. Test 1 is thus designed to find out how the different degrees of dynamism affect the planning results using *RHP-INT*.

10.1.1 Measuring dynamism of instances

Dynamism of instances can be measured by specific indexes. In Lund et al. (1996), a very simple measurement defined as the *degree of dynamism* (DOD) is proposed, which is calculated as the ratio between the number of dynamic requests and the total number of requests. Larsen (2001) also considers the release time of the requests and proposes the *effective degree of dynamism* (EDOD) δ^e. Given the total length of the entire time

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_10,

horizon T, δ^e is defined as follows:

$$\delta^e = \frac{1}{n}\sum_{r\in R}\frac{t_r^{rls}}{T} \tag{10.1}$$

Considering the time windows related to the requests, Larsen (2001) also introduces an extension to the EDOD (EEDOD) δ^e_{TW}:

$$\delta^e_{TW} = \frac{1}{n}\sum_{r\in R}(1 - \frac{e_r - t_r^{rls}}{T}) \tag{10.2}$$

e_r is the end of the time window associated with request r. Although this index is proposed rather for the dynamic VRPTW, it can be adjusted easily for the DCFPDPF considered here by substituting e_r with e_{r^+}, which is the end of the time window of the pickup operation of request r. As the FTL transportation is studied here, there is no further need to take the time window of the delivery operation into account. Thus, the EEDOD for the DCFPDPF is given by:

$$\delta^e_{TW} = \frac{1}{n}\sum_{r\in R}(1 - \frac{e_{r^+} - t_r^{rls}}{T}) \tag{10.3}$$

10.1.2 Instance generation

In order to conduct the computational study, some new test instances with the same request information but different DOD have to be generated. In the first step, ten static instances are generated in total. The procedure of generating static instances begins with generating request information in an iterative fashion. The length per planning period is set as $\tau = 100$. In each iteration it, $it = 1, \ldots, 30$, which corresponds to the time interval $(\tau \cdot it, \tau \cdot it + \tau]$, about 40 requests for the entire coalition are generated. In order to capture fluctuation of customer demands, this number is adjusted randomly with an amount up to $\pm 20\%$. These requests are then assigned to three coalition forwarders according to the request weights, which are also randomly generated in $[0.7, 1.3]$ for each forwarder in each iteration. As a result, each instance consists of three forwarders and on average 1,186 requests over the entire time horizon.

Pickup and delivery locations of requests are randomly distributed in a rectangle of size 200×150. The distance between any two location nodes is the Euclidean distance. For more than 80% of all requests, the distance between pickup and delivery locations lies in the range from 20 to 160 and the average value is about 90. The time windows for a request r generated in iteration it are defined in the following way. Let r^+ and r^-

represent the pickup and delivery locations of request r. In the first iteration ($it = 1$), b_{r^+} is given a random value in $[\tau/3, \tau]$. In the following iterations, b_{r^+} is given a random value in range $(\tau \cdot it, \tau \cdot it + \tau]$. e_{r^+} is calculated by adding b_{r^+} with a time window width, which is determined as $\tau/2 \pm 30\%$. The time window for the delivery location $[b_{r^-}, e_{r^-}]$ is simply defined as $[b_{r^+} + d'_{r^+r^-} + s_{r^+}, e_{r^+} + d'_{r^+r^-} + s_{r^+}]$, while $d'_{r^+r^-}$ is the driving time from r^+ to r^- and s_{r^+} is the service time at r^+. All operations are assigned the same service time of 10. Since the execution of some requests generated in the last iteration $it = 30$ may be finished later, the entire time horizon of the instances is extended to $[0, 3300]$.

Each forwarder is assigned a vehicle fleet. The number of vehicles is determined as the average request number per planning period with a deviation of up to $\pm 30\%$. Vehicles are located at randomly generated start locations with empty load at the very beginning t_0. The average number of vehicles per forwarder in an instance is 13.3, while the concrete numbers are varying from 9 to 17. Fixed costs of vehicles α_k are supposed to be the same by all forwarders so that they can be ignored in the computational study. The velocity of all vehicles is assumed to be one so that the driving time between two nodes equals the distance. The variable cost rate β_k for a DU is set to one for all vehicles $k \in K$.

Since requests are allowed to be transferred to common carriers, the price for outsourcing requests must also be specified. This cost γ_r for a request r is calculated as $\gamma_r = \varphi d_r \theta^{d_r}$, where φ is a constant cost rate per DU and d_r is the adjusted travel distance between pickup and delivery locations. The basic cost rate φ is given the value of 2 that is twice as high as the variable cost rate of the vehicles β_k and the adjusted travel distance is defined as $d_r = \max\{5, d_{r^+r^-}\}$. The motivation to use the adjusted travel distance is that the common carriers charge a fixed minimum fee for each request if the distance to travel lies below a specific level. θ is a parameter which is set to 0.9986. θ^{d_r} can be seen as a distance-dependent discount on the cost rate. The introduction of θ captures the fact in practice that freight rates reduce with increasing transportation distance.

The second step is to assign each request r a release time t_r^{rls} to make a static instance to a dynamic one. $\delta_1\%$ of all requests are given the release time of $(b_{r^+} - 3\tau)$, $\delta_2\%$ of $(b_{r^+} - 2\tau)$, and $\delta_3\%$ of $(b_{r^+} - \tau)$. Negative values are adjusted to zero since the simulation starts at $t_0 = 0$. Through changing the values of δ, several dynamic instances with different degrees of dynamism can be generated based on a single static instance generated in the last step. Using the ten static instances, three sets of DCFPDPF instances are generated. They have different degrees of dynamism: strongly dynamic (SD), moderately dynamic (MD), and weakly dynamic (WD). The parameter triple $(\delta_1, \delta_2, \delta_3)$ is fixed for set SD to $(10, 10, 80)$, for MD to $(10, 80, 10)$, and for WD to $(80, 10, 10)$. As a result, the average EEDOD of the sets SD, MD, and WD are 0.945, 0.924, and 0.904, respectively. Even for set WD where the most requests are released at least three planning periods ahead, the

EEDOD is still very high due to the extremely long horizon.

10.1.3 Simulation results

Both scenarios IP and CTP are simulated in this computational study. An important issue in the simulation is the specification of due requests. In this study, a known request is declared for planning period $p+1$ at time $t = p\tau$ as a due request if its pickup service must be started before $(p+1.25)\tau$. For the very last period, all requests are labeled as due requests. Both strategies MYP and FLP are considered for the planning scenarios IP and CTP. As a result, four settings are defined for each instance set: (IP, MYP), (IP, FLP), (CTP, MYP), and (CTP, FLP). While using the FLP strategy, the potential outsourcing costs are adjusted by a weight smaller than one. Non-due requests with $(p+1.25)\tau < e_{r^+} \leq (p+2)\tau$ are multiplied with a weight factor of 0.75, while the remaining non-due requests with $e_{r^+} > (p+2)\tau$ are multiplied with a factor of $0.75^2 \approx 0.56$.

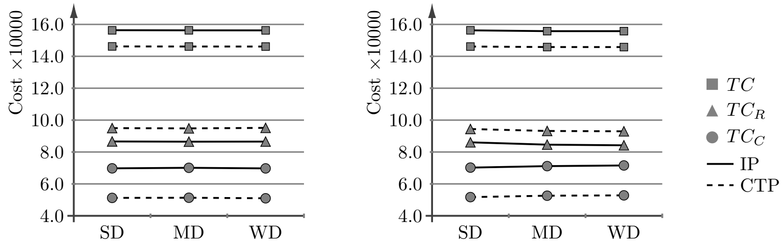
For each instance and simulation setting, the simulation has run three times and the average costs per instance are calculated. The average costs of all ten instances in each set and the average number of outsourced requests of the entire coalition n_C are given in Table 10.1.

Table 10.1: *Results of Test 1*

	SD		MD		WD	
	TC	n_C	TC	n_C	TC	n_C
IP,MYP	156375	541	156287	545	156262	543
IP,FLP	156290	544	155767	554	155767	558
CTP,MYP	146197	434	146137	436	146120	433
CTP,FLP	146134	441	145805	455	145791	456

On the one hand, CTP obviously outperforms IP for both MYP and FLP and all instances. The cost reduction realized by CTP roughly amounts to 6.5%. On the other hand, forwarders can reduce costs by planning in a forward-looking way. However, compared with CTP, the benefits of FLP are rather ignorable. It can be concluded that it is a much more promising strategy to perform CTP than just trying to improve the own planning individually.

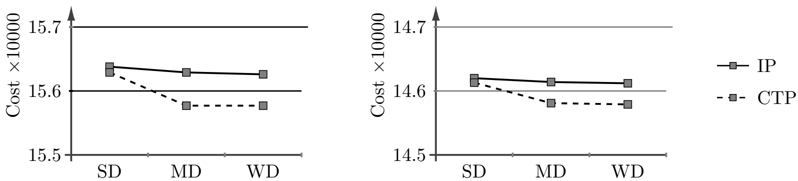
The reason of the cost-savings through CTP can be derived from a comparison of the total number of outsourced requests n_C given in Table 10.1 and the composition of the total costs TC shown in Figure 10.1. Besides TC, Figure 10.1 also shows the total route costs TC_R and the total costs of subcontracting requests to common carriers TC_C. The costs of routes and the number of requests planned in routes of CTP are significantly higher than those of IP, while the total costs TC are lower. This indicates that the routes constructed in CTP are obviously more efficient and the decisions on outsourcing are made in a better way.



(a) MYP **(b)** FLP

Figure 10.1: *Cost composition of the results of Test 1*

Comparing the performance of CTP for different sets, CTP performs even slightly better for the strongly dynamic set SD than for the other two sets MD and WD. This implies that CTP is more valuable for forwarders when a quick response to the release of customer requests is expected because the exchange of requests increases the possibility to find a cheaper way to fulfill the requests and this is very important in highly dynamic environments.



(a) Isolated planning **(b)** Collaborative planning

Figure 10.2: *Comparison between myopic and forward-looking planning (Wang and Kopfer, 2013a)*

It can be observed that if forwarders plan myopically, the quality of the planning suffers little from the increment of dynamism of the instances. However, the benefit of having request information in advance can be clearly seen when forwarders plan in a forward-looking way. For set SD, FLP performs almost equally with MYP because there is little information known about the future. For sets MD and WD, FLP clearly outperforms MYP. This implies that forwarders are able to reduce costs if they can acquire request information in advance and consider it in their planning. It can be seen from Figure 10.1b that the reduction of the total costs with an increasing number of requests being released earlier can be attributed to the decreasing total route costs TC_R, although the total number of outsourced requests n_C and their costs TC_C increase slightly. This observation indicates that the decision between self-fulfillment and subcontracting can be made better when forwarders are given more time for planning. However, the solution quality cannot be enormously improved by having this information much earlier than necessary. This effect can be seen through the comparison of the results for sets MD and WD in Figure 10.2. In other words, in a dynamic environment, it is unnecessary for forwarders to consider such requests in their current planning that are to be fulfilled in the far future.

10.2 Test 2: Value of advanced request information

An important observation on the results of Test 1 is that even when forwarders are offered the information about the requests that are to be fulfilled in the far future, they do not need to consider these requests in the planning immediately. In the framework *RHP-INT*, this differentiation of requests according to their urgency can be done by choosing a proper weight function (see Section 9.3.4). Test 2 is thus designed to answer the question, what this function should be.

In this test, *RHP-INT* is used for solving the DCFPDPF. Three different planning strategies are tested: The first strategy is the MYP, which totally ignores the requests that are not urgent. The second strategy refers to the consideration of all known requests, which is denoted as FLP-I. The last strategy is an FLP that only considers future requests to a limited extent and is denoted as FLP-II.

10.2.1 Instance generation

In order to test the three different planning strategies, ten larger instances than those used in Test 1 are generated using the same methods as described in Section 10.1.2. The entire time horizon is extended from [0,3300] to [0,4800] by generating instances with 45 instead of only 30 planning periods à 100 TU. The most important change in the parameter settings concerns the distribution of the release time. Denote the *time span* between the release time t_r^{rls} of a request r and the beginning of the time window of the corresponding pickup operation b_{r+} as $t_r^{ts} = b_{r+} - t_r^{rls}$, the new instances are generated in such a way that 5% of the requests have $t_r^{ts} = 450$, 10% $t_r^{ts} = 400$, 10% $t_r^{ts} = 350$, 15% $t_r^{ts} = 300$, 30% $t_r^{ts} = 250$, 20% $t_r^{ts} = 200$, and 10% $t_r^{ts} = 150$. The EEDOD of the ten new instances is 0.9349. They will also be used in the following three tests.

10.2.2 Simulation settings

The most simulation settings of this test are the same as those of Test 1. The strategy of MYP can be realized by assigning due requests the weight of one and non-due requests of zero. FLP-I uses the same settings as FLP in Test 1. The weight of due requests is one. Non-due requests with $(p+1.25)\tau < e_{r^+} \leq (p+2)\tau$ have a weight of 0.75 and all other non-due requests have a weight of 0.56. The weight distribution in the FLP-II is specified as follows. Due requests that must be served in $(p+1.25)\tau$ have a weight of 1. Non-due requests with $(p+1.25)\tau < e_{r^+} \leq (p+2)\tau$ have a weight of 0.75. Non-due requests with $(p+2)\tau < e_{r^+} \leq (p+2.5)\tau$ have a weight of 0.56. The remaining known requests are ignored and assigned a weight of zero. As the length of each planning period τ equals 100, the lengths of the planning horizon of each static planning of the three strategies MYP, FLP-I, and FLP-II are then 125, ∞, and 250, respectively.

10.2.3 Results and discussion

In order to identify the cost-savings that can be realized by using *RHP-INT*, the scenarios IP and CTP are simulated. Each instance has been solved three times. The average values of the results obtained in the three trials of all ten instances are summed up and given in Table 10.2.

Table 10.2: *Results of Test 2*

	MYP	FLP-I	FLP-II
IP	236199.37	235382.59	235401.93
CTP	220956.44	220703.65	220571.02
ΔC_{CTP}	15210.56	14714.79	14830.91
$\Delta C_{CTP}(\%)$	6.44	6.25	6.30

The total costs of the plan over the entire time horizon of the coalition with and without cooperation are given in the second and third rows. It is clear that CTP outperforms IP with all planning strategies from MYP to FLP-II. The realized cost-savings are given in the fourth row in absolute value and in the fifth row in percentage. In all cases, CTP can realize over 6% cost-savings compared with the results without horizontal cooperation. The best average result of IP is achieved by using the planning strategy FLP-I and of CTP by using FLP-II. From a further comparison of the total costs between FLP-I and FLP-II it can be concluded that FLP-II performs equally well as the more time-consuming strategy FLP-I. This result confirms the hypothesis made based on the results of Test 1 that requests to be fulfilled in the far future do not need to be considered in the current planning process to get good quality solutions.

Table 10.3: *Cost-savings through forward-looking planning*

	FLP-I		FLP-II	
	ΔC_{FLP}	$\Delta C_{FLP}(\%)$	ΔC_{FLP}	$\Delta C_{FLP}(\%)$
IP	816.78	0.35	797.44	0.34
CTP	252.79	0.11	385.42	0.17

Table 10.3 shows the cost-savings of FLP compared to MYP. The cost-savings in columns two and four are calculated by subcontracting the total costs resulted in FLP-I and FLP-I from the total costs of the results of MYP. The relative values of the cost-savings are then given in columns three and five. It is clear that in both IP and CTP, FLP can lead to better solutions. Especially, FLP can realize more percentage of cost-savings in IP than in CTP. However, if we compare the cost-savings realized by FLP with those by CTP, it can be concluded that collaboration has more positive impact than FLP. Even the worst solution of CTP (obtained by MYP) is much better than the best solution of IP (obtained by FLP-I) and the cost-savings achievable by introducing collaboration is over 18 times than that achievable by applying FLP (6.44% vs. 0.35%). It is worth mentioning here that this comparison does not indicate a conquer between the CTP and FLP. In fact, the best results are achieved by conjointly applying both of these two strategies.

Based on the discussion about the results of Test 2, some important conclusions can be drawn for freight forwarders. Advanced request information is valuable and has to be considered. However, requests that are to be fulfilled in the far future can be ignored without necessarily worsening the overall results. For a single forwarder, it is recommendable to use request information offered in advance to reduce the total costs. However, CTP is obviously more efficient in reducing costs. Compared with cooperation, the benefits of improving individual planning strategy become negligible and this finding coincides with that of Test 1. Forwarders should seek for cooperation with proper partners for more potential of cost reduction.

Another interesting observation is that the cost-savings through CTP using all three planning strategies are on the same level. It seems that for a given configuration of *RHP-INT*, the reduced costs through CTP against IP remain stable. The next two tests are then designed to exam this hypothesis by varying the length of the planning horizon L_H (in case of *RHP-RT*) and the length of the planning period τ (in case of *RHP-INT*) of the rolling horizon planning and to find out good values of these important parameters. Since FLP-II performs equally well as FLP-I but requires significantly less computational efforts, the forward-looking planning strategy FLP-II is used in the following tests.

10.3 Test 3: Length of the planning period in *RHP-INT*

Test 3 is designed to identify the best length of the planning period in *RHP-INT*, given such a weight function that is described in the last section for FLP-II. According to this function, the length of the planning period is 2.5τ, as requests with e_{r^+} later than $t+2.5\tau$ will be excluded from the planning at time t.

10.3.1 Simulation settings

In this test, six different τ-values: 25, 50, 75, 100, 125, and 150 are tested. Since the entire time horizon is fixed, the number of the total planning periods can be calculated simply by $4500/\tau + 1$. Note that the last 300 TU in the entire time horizon are reserved to make sure that the service of all requests can be finished and the last planning period is defined as [4500,4800] for all τ-values. For instance, by setting $\tau = 25$, the entire time horizon is divided into 181 planning periods and the first plan is made at time $t = 0$ and the last one is made at $t = 4500$. For the largest τ-value of 150, there are only 31 planning periods.

In order to determine the cost-savings that can be realized by using the proposed approach, the two scenarios IP and CTP are simulated in this test. Each instance has been tested three times in both scenarios and the average values of these three trials are calculated for each instance.

10.3.2 Results and discussion

The results of the simulation are given in an aggregated form in Table 10.4. The costs given in the second and third rows are the average values of the ten instances. The last two rows give the realized cost-savings (absolute and relative) through CTP.

Table 10.4: *Results of total costs of Test 3*

τ	25	50	75	100	125	150
IP	240997.53	236092.67	236446.13	235401.93	235886.31	236949.87
CTP	226821.20	220726.27	221809.72	220571.02	221722.27	223548.80
ΔC	14176.33	15366.40	14636.41	14830.91	14164.04	13401.07
$\Delta C(\%)$	5.88	6.51	6.19	6.30	6.00	5.66

An illustrative comparison of the results reported in Table 10.4 among the settings is given by Figure 10.3. Two interesting characters can be observed from this figure: both the two curves depicted in this figure have a “W”-shape and they are nearly parallel to each other.

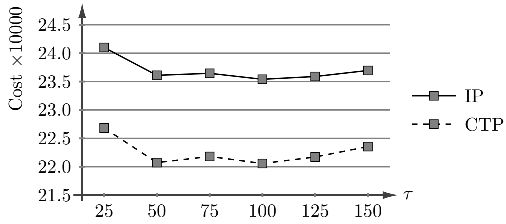


Figure 10.3: *Performance of* RHP-INT *with different τ-values*

10.3.2.1 Total costs of the solutions

It is not surprising that a too small or a too large τ-value leads to inferior solutions than a well-chosen moderate value. On the one hand, very small τ-values result in high frequency of the planning and short planning horizons. For $\tau = 25$, the length of the planning horizon is only $L_H = 2.5 \times 25 = 62.5$, which is even shorter than the time corresponding to the average transportation length $\bar{d}' = (\sum_{r \in R} d'_{r^+ r^-})/|R| = 93$ of the requests. In this extreme situation, vehicles are only assigned one due request each time and no successive request will be considered, since the time when the service will be finished at the delivery location of the assigned due request lies beyond the planning horizon and the possible successive requests will be ignored in the planning.

On the other hand, if τ is given a large value, the planning process becomes unresponsive to the dynamic of the instances. When $\tau = 150$, the length of the planning horizon L_H is 375, which is much longer than $\bar{d}'$. In this case, the planning process tries to fix the plan for too many requests so that the resulted plans are too rigid to be competitive with those obtained by configuring the *RHP-INT* with moderate τ-values.

In order to better understand the two valleys of the curves, it is helpful to take a glance at the composition of the total costs shown in Figure 10.4.

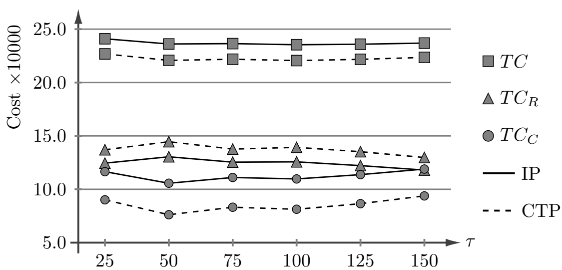


Figure 10.4: *Cost composition of the results of Test 3*

Since subcontracting is introduced in the DCFPDPF, the quality of the results depends not only on a good routing decision, but also on a good decision on the fulfillment modes for each single request. The basic principle used in the mode selection is to compare the insertion cost Δc_r of a request r (see Section 2.4.3) and the outsourcing price γ_r. In this problem, Δc_r depends on both the distance between its pickup and delivery locations $d_{r^+r^-}$ and the distances to the previous and successive nodes in the route that will be connected with r^+ and r^-.

Generally speaking, the shorter the planning period is, the fewer requests are considered in the planning. In case of shorter planning periods, there exist few opportunities to bundle requests and to generate efficiency routes. As a result, the repositioning costs caused by sending vehicles from one request to another also tend to be higher than in planning with longer planning period. The average insertion cost also increases with decreasing number of requests to be planned each time. On the contrary, if more requests are to be inserted into a route due to a longer planning period, the possibility increases that they can be well bundled so that better routes can be found. The distance from a delivery location to the pickup location of the next request also tends to be shorter. In other words, the shorter the planning period is, the more expensive it is on average to insert requests into routes and the higher the possibility is to outsource requests to common carriers. This can be intuitively understood as such that the planning prefers subcontracting against self-fulfillment in the mode selection with small τ-values. This is the reason why the outsourcing costs of the plan with $\tau = 25$ are by far the higher than other cases that can be seen in Figure 10.4.

With increasing length of planning periods, there are more and more requests that can be well bundled in each planning and the routes become longer and longer. The preference of the planning process on outsourcing exists with increasing τ-values until some time when the insertion costs are reduced to a level that is slightly lower than the outsourcing costs. The reduction of the insertion costs in general leads to the change of the preference on fulfillment mode from subcontracting to the coalition's vehicles. This point is reached in this test at $\tau = 50$. The new strong preference of mode on vehicles makes the approach try to put requests into vehicle routes as many as possible. The result is the highest level of route costs and the lowest level of the outsourcing costs. In total, the overall results are the second best in both IP and CTP scenarios.

When τ further increases, the relationship between the meaning of the due requests and non-due requests in the planning changes. A small τ-value indicates a short planning horizon. The majority of the time in each planning period is used merely for fulfilling the due requests. The candidate routes generated in the planning consist of almost only the due requests. This means that due requests are emphasized much more than non-due requests. In a planning with large τ-values, i.e., long planning horizons, longer routes are generated, in which after the due requests also non-due requests are planned. The non-due requests to be planned in the next planning periods are thus emphasized more than in planning processes with small τ-values. In other words, small τ-values emphasize the near future and large τ-values emphasize a more smooth plan over a longer time. A clear preference with a proper configuration, either better plans for the near future ($\tau = 50$) or smooth plans ($\tau = 100$) can help find good solutions in general, while the configuration $\tau = 75$ with an unclear preference leads to worse results. But it must be mentioned here that that $\tau = 75$ means an unclear preference only refers to the average performance. As a matter of fact, this configuration leads to clear emphasis and in turn the best results of three instances.

10.3.2.2 Cost-savings through CTP

The benefits of CTP can be clearly seen from Figure 10.3. On the one hand, CTP obviously outperforms IP in all cases and has realized considerable cost-savings that account to 5.88%-6.51%. Moreover, the two curves of the total costs are almost parallel. It indicates that the performance of the CTP is very stable against the choice of the τ-value and CTP is always about 6% better than without request exchange. For a single freight forwarder this means that (1) he can improve his planning by choosing the proper configuration of the rolling horizon approach, and (2) he can always achieve a further noticeable improvement by collaborative request exchange while the improvement remains relatively constant no matter what configuration has been chosen.

On the other hand, the cost-savings with large τ-values are generally less than with small values. It can be explained by the fact that in planning with short planning horizons, forwarders can hardly construct efficient routes in the IP scenario. In the CTP scenario, however, the coalition has a larger request pool as well as a larger fleet so that request exchange among the members can considerably improve the results. On the contrary, a long planning horizon means at the same time a large number of requests to be planned in each planning period and forwarders can better bundle their requests in the IP scenario, too. As a result, the potential of further improvement of the routes by CTP decreases a little. This observation also indicates an important conclusion that in highly dynamic environments that require quick responses of forwarders and thus short planning periods, CTP tends to be even more useful to reduce costs than in static planning.

10.4 Test 4: Length of the planning horizon in *RHP-RT*

Test 4 is designed in the same way as Test 3 to analyze the influence of different lengths of the planning horizon configured in the planning framework *RHP-RT* on the performance of the rolling horizon approach.

10.4.1 Simulation settings

In the planning framework *RHP-RT* there is no planning period defined. However, in order to enable a fair performance comparison between *RHP-INT* and *RHP-RT*, fictive planning periods which have the same durations as those tested in Test 3 need to be introduced in this test. Using the same weight function the two approaches will have the same length of the planning horizon L_H when the same τ-value is given.

Different to *RHP-INT*, the number of plannings of the *RHP-RT* does not depend on L_H but on the number of requests in total. The average number of requests of the ten instances is 1,775. As some requests have the same due time and all requests whose due time after $t = 4500$ are planned at once in the last planning, the number of actually performed plannings in the entire time horizon is somehow less than the number of requests and accounts to 1,462 on average. This number also implies that the computational efforts needed for the whole planning increase considerably.

Again, the two scenarios IP and CTP are simulated for each instance. Three trials of simulation ran for each instance and the average value is calculated.

10.4.2 Results and discussion

The results of the simulation are given in Table 10.5 in the same format as in Table 10.4 and illustratively plotted in Figure 10.5.

Table 10.5: *Results of total costs of Test 4*

$L_H(\tau)$	63(25)	125(50)	188(75)	250(100)	313(125)	375(150)
IP	243523.00	238745.87	238987.03	238105.87	238396.85	239788.12
CTP	228592.63	223371.96	224721.87	223334.53	224263.84	225244.73
ΔC	14930.37	15373.91	14265.16	14771.34	14133.01	14543.39
$\Delta C(\%)$	6.13	6.44	5.97	6.20	5.93	6.07

10.4.2.1 Total costs of the solutions

The results of Test 3 reconfirm that CTP is superior to IP. The realized cost-savings account to more than 6% on average. Also in this test, even the worst CTP solution

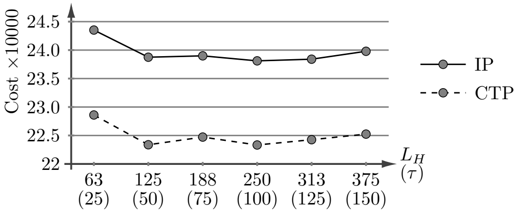


Figure 10.5: *Performance of* RHP-RT *with different* L_H*-values*

is substantially superior to the best IP solution. The curves of costs of IP and CTP in Figure 10.5 have a "W"-shape similar to those in Figure 10.3. The overall best result is achieved by setting L_H to 250, which is the same as in Test 3 with $\tau = 100$. Tests 3 and 4 show that the best choice for the length of the planning horizon L_H seems to be independent on the planning framework.

10.4.2.2 Comparison between *RHP-INT* and *RHP-RT*

A straightforward comparison between the performances of the two rolling horizon planning approaches is given in Figure 10.6. *RHP-INT* is clearly superior to *RHP-RT* in both IP and CTP scenarios. The extremely high frequency of the change of existing plans over the entire time horizon makes the planning framework of the latter one over sensitive to the dynamic of the instances.

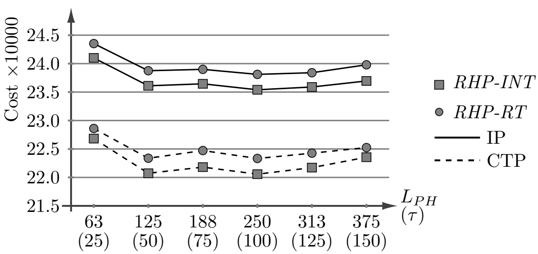


Figure 10.6: *Comparison between* RHP-INT *and* RHP-RT

Another interesting observation is that although using different frameworks can result in different levels of solution quality, all four curves drawn in Figure 10.6 share the same shape. Moreover, for the tested configurations, they are almost parallel. This means that the choice of the planning approach in a dynamic environment has a strong influence on solution quality. On the contrary, the relative performance of different configurations of a specific rolling horizon planning approach remains stable. The settings of the rolling horizon planning should be chosen based on instance characters and are valid for both approaches. A practical interpretation is that the individual experience on solving dynamic routing problems can also be well used in CTP.

10.5 Test 5: Planning with high subcontracting costs

The last test in this computational study on the DCFPDPF is designed to observe the consequences of a dramatic increment of the price level of freight charges paid to common carriers for subcontracting. The planning approach based on *RHP-INT* is chosen for this test due to its superior performance against *RHP-RT*.

10.5.1 Instance adjustment and simulation settings

The same instances tested previously in Tests 2-4 are used in this test. However, the subcontracting costs of the requests are increased by 50%. The rolling horizon planning is configured the same as that used in Test 3.

Due to the cost function used for generating the subcontracting price for requests in the instance generation (see Section 10.1.2), subcontracting becomes so expensive in this test that for a given request r with the travel distance $d_{r^+r^-}$ the freight charges of outsourcing can cover almost the vehicle costs for traveling a distance that is three times as long as $d_{r^+r^-}$. Such extremely high costs of subcontracting force the planning to avoid any outsourcing whenever the capacity of the fleet is not exhausted.

10.5.2 Results and discussion

Results of the simulation are given in Table 10.6 in the same format as in other tests and are illustratively shown in Figure 10.7.

Table 10.6: *Results of total costs of Test 5*

τ	25	50	75	100	125	150
IP	279565.80	269478.55	289319.15	287498.22	290319.97	293857.36
CTP	258279.35	247962.40	266892.00	265388.23	269454.49	274699.00
ΔC	21286.45	21516.15	22427.15	22109.98	20865.48	19158.35
$\Delta C(\%)$	7.61	7.98	7.75	7.69	7.19	6.52

The absolute realized cost-savings through CTP increase significantly compared to Test 3, because the whole coalition does not have enough capacity to fulfill all requests and thus have to outsource requests even for the higher cost level. Even more important, the relative cost-savings given in the fifth row in Table 10.6 indicate that the more expensive the subcontracting is, the more significant the synergy effect of collaboration is.

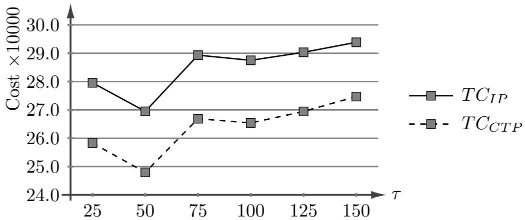


Figure 10.7: *Performance of* RHP-INT *with high subcontracting costs*

Since subcontracting is considered in the DCFPDPF, a feasible solution to the problem is to transfer all requests to common carriers. Due to the extremely high costs of subcontracting, the objective of the problem in this test is almost equivalent to that of reducing the costs for subcontracting to the greatest extent by selecting and inserting requests into routes. The benefits of inserting an FLT request r into a vehicle route in the DCFPDPF can be calculated by substituting the increment of the route costs from the outsourcing price. The latter one can be calculated by multiplying the variable cost rate of the vehicle β with the sum of the repositioning distance of the vehicle from its current location to the pickup location of the request d_r^{repo} and the distance for transporting the goods to the delivery location $d_{r^+r^-}$. Then, an index η_r can be introduced to measure the efficiency in reducing costs of the insertion of a request r into some vehicle route. η_r is defined in (10.4) as the ratio between the cost reduction of inserting r into some vehicle route and the non-effective driven distances, i.e., how much cost can be reduced by each DU of repositioning of the vehicle for this request:

$$\eta_r = \frac{\varphi d_r \theta^{d_r} - \beta(d_{r^+r^-} + d_r^{repo})}{d_r^{repo}} \tag{10.4}$$

When $d_{r^+r^-} \geq 5$, which is the absolute majority in R, d_r and $d_{r^+r^-}$ are the same because $d_r = \max\{5, d_{r^+r^-}\}$. We can then replace $d_{r^+r^-}$ with d_r in (10.4) and take a deeper look at the relationship between d_r and η_r. The partial derivative of η_r with respect to d_r, $d_r \geq 5$ is:

$$\frac{\partial \eta_r}{\partial d_r} = \frac{1}{d_r^{repo}} \left[\varphi \theta^{d_r}(1 + d_r \ln \theta) - \beta\right] \tag{10.5}$$

The second order partial derivative of η with respect to d_r is:

$$\frac{\partial^2 \eta_r}{\partial d_r^2} = \frac{\varphi \ln \theta}{d_r^{repo}}(2\theta^{d_r} + d_r \theta^{d_r} \ln \theta) \tag{10.6}$$

Since $\theta = 0.9986$ (see Section 10.1.2), we have $\ln 0.9986 = -0.0014$. As the maximal length of a single request is the diagonal of the rectangle in which all customer nodes are located, $d_r \leq 250$. The term $2\theta^{d_r} + d_r\theta^{d_r} \ln \theta$ is always positive when d_r takes values in the range of [5,250]. Thus, the second order partial derivative of η_r with respect to d_r (10.6) is always negative, which means that the first order derivative (10.5) is a monotone decreasing function of d_r. The first order derivative of η_r at $d_r = 250$ can be calculated using the given values: $\varphi = 3$, $\theta = 0.9986$, $\beta = 1$, and $d_r^{repo} > 0$:

$$\left.\frac{\partial \eta_r}{\partial d_r}\right|_{d_r=250} = \frac{1}{d_r^{repo}}\left[3 \cdot 0.9986^{250}(1 + 250 \cdot \ln 0.9986) - 1\right] = \frac{0.3733}{d_r^{repo}} > 0$$

The positive value of the first order derivative of η_r indicates that η_r defined in (10.4) is a monotone increasing function of d_r for a given d_r^{repo}. For each DU driven for the reposition a vehicle to a request, the longer the request is, the more efficient this insertion is. In other words, The more the long requests are subcontracted, the worse the quality of the planning tends to be.

This idea can then be used for a better understanding of the results of this test. A good solution should have two characters. The first one is a good mode decision and the second one is a high efficiency of the vehicle routes. Based on the analysis above, the first character can be quantified approximately using the average request length of outsourced requests $\bar{d_r^C}$ of the solution. In general, the smaller this value is, the more long requests are planned in vehicle routes and thus the quality of the solution tends to be high. The second character can be quantified by the efficiency of the vehicle fleet η_K, which is defined as the ratio of total driven distances for repositioning vehicles to the total route lengths of all vehicles in K, where K is the entire vehicle set of the coalition. Table 10.7 gives these two indexes of CTP solutions.

Table 10.7: *Characters of CTP solutions of Test 5*

τ	25	50	75	100	125	150
η_K	0.134	0.135	0.173	0.175	0.171	0.173
$\bar{d_r^C}$	60.3	55.2	62.9	62.3	62.4	63.7

When τ=25, the whole planning is quite greedy compared with other configurations. Requests with both long and short distances d_r will be inserted into any vehicle routes each time so long as the capacity restriction is satisfied without considering what happens afterward. Compared with the best result at $\tau = 50$, this myopic behavior results in a high $\bar{d_r^C}$-value, which indicates that more requests with longer transport distances d_r are outsourced, even when the routes are the same efficient.

When τ=50, more attention is paid to the routing aspect in the dynamic problem than the previous case. On the one hand, the planning process still tries to put every request into vehicle route. On the other hand, it also tries to pick up the requests with longer transport distances, even when they are slightly less urgent than some shorter requests. This can be interpreted from the character indexes. The efficiency of vehicle routes is only one pro mile worsen than that when τ=25, but the mode choice is made in a significantly better way so that the total performance becomes much better.

The unclear preference on greedy plans or smooth plans at τ=75 also leads in this test to obvious worsening of the solution quality compared to those obtained with τ=50. Although the results can be improved somehow by increasing τ from 75 to 100, due to the increasing length of the planning period the whole planning process is no more flexible enough to deal with the dynamic of the instances. The intent to reduce great changes in the partial plans for the non-due requests in the following planning periods weakens the capability of the approach to make enough quick responses to new released requests, especially when the distance between the pickup and delivery locations of these requests is long.

Finally, it is necessary to emphasize that the analysis of the results referring to the two indexes η_K and $\bar{d_r^C}$ are based on an extreme situation and thus cannot be simply generalized for other tests. In this test, the price level of outsourcing freight charges is extremely high so that it is almost always profitable to fulfill a request with the vehicle fleet rather than outsourcing, even without considering the synergy effect of bundling requests. Precisely, the break even point for a single request can be calculated for a given d_r^{repo}. We can take the CTP results of this test as an example and use the average repositioning distances per request for the calculation. The average value of d_r^{repo} over all configurations is 21.70 and the break even point lies at $d_r = 11.11$. Consequently, so long as the transportation distance between the pickup and delivery locations of a request is longer than 11.11 DU, the fulfillment using coalition's own vehicle is better. For comparison, using the same average repositioning distance, the break even point for the outsourcing price level assumed in the previous tests would be at $d_r = 23.19$, which is even longer than the repositioning distance. In this case, the strategy of preferring self-fulfillment against subcontracting by predominantly inserting any request into routes will not be valid for very small τ-values. This is demonstrated by the fact that for the configuration with $\tau = 25$ the average numbers of outsourced requests of all ten instances in Tests 3 (675) and 4 (696) are significantly smaller than that in this test (714).

10.6 Conclusions

In this chapter, a comprehensive computational study on the dynamic CTP is conducted. It begins with the tests reported in Wang and Kopfer (2013a), which is designed to test the performance of the solution approach developed based on the rolling horizon planning framework *RHP-INT*. Results of this test show a clear success of the proposed approach in reducing costs of horizontal coalitions of forwarders. By varying the degree of dynamism of the instances, it comes out that if forwarders can get the request information in advance, they can improve their planning and reduce costs. However, they do not need to consider all request information in a forward-looking way, but can ignore the requests that are to be fulfilled in the far future. Notice that the phrase “far future” here has to be understood according to the configuration of the rolling horizon planning and refers to the time subsequent to the next few planning periods.

For the next four new designed tests, a new set of instances with longer entire time horizon and more requests are generated. In Test 2, different planning strategies using the request information released by customers in advance are tested. The results offer strong supports for the suggestion derived from Test 1.

In the subsequent two tests, the two rolling horizon planning approaches proposed in Chapter 9 are tested with different configurations with respect to the length of the planning horizon. Results show that the *RHP-INT* outperforms *RHP-RT* in terms of both solution quality and computational efforts. The two approaches perform differently in terms of solution quality with different configurations. The choice of the right configuration significantly affects the results. Freight forwarders can thus improve their planning technique to achieve better business performance. However, the more promising way to reduce their operational costs is to seek for cooperation and do CTP by exchanging requests. Results show that the coalition can always expect the similar amount of cost-savings through CTP with any tested configuration. It implies that the CTP has no conflict with the improvement of the planning techniques in reducing costs. Individual planning settings that have proved to be successful can be used for deriving proper configurations of the CTP in coalitions.

In the last test the costs of the services of common carriers are increased to an extremely high level. In this situation, a higher flexibility of the solution approach to actualize the previous plans becomes the key factor for the success of the rolling horizon planning approach and a generally smaller τ-value should be chosen. Furthermore, the realized cost-savings are higher than those of Tests 3 and 4. This phenomenon implies that collaboration can better compensate the increasing prices of common carriers.

11 Conclusions and future research

In this thesis, the operational transportation planning of modern freight forwarding companies is studied. The motivation of this study is that the forwarders are confronted with increasing pressure to increase their operational efficiency of request fulfillment while the internal cost-saving potential is almost exhausted. A further cost reduction can be realized by taking advantage of the synergy effect embedded in cooperation with other forwarders. The extended spectrum of options for request fulfillment offered by cooperation results in more complicated optimization problems in the operational transportation planning. The purpose of the research in this thesis is to develop efficient solution approaches that can help forwarders master the high complexity of these planning problems in the new situation.

11.1 Summary and conclusions

In order to study the complicated optimization problems in the operational transportation planning that arise in different forms of cooperation, the basic decision problems of forwarders are introduced in the first part of this thesis. The first type of these problem, i.e., the classical vehicle routing and scheduling problems are described in Chapter 2. These problems have to be solved when only the own vehicles are used for the fulfillment of customer requests under the assumption that there exist sufficient capacities in the vehicle fleet. In the last several decades, a great number of specific problems have been proposed and discussed in literature in this research field. Instead of trying to give a comprehensive overview of all of them, only the very fundamental problem types are introduced in this chapter. The focus lies in the PDPTW which is the underlying routing problem in the scenarios of IOTP and CTP discussed in the later chapters of this thesis. Moreover, the ALNS heuristic proposed by Ropke and Pisinger (2006) is described in the last section of this chapter. This heuristic is then further developed and used as the solution approach for the routing problems in the IOTP and CTP scenarios.

The second optimization problem that has to be solved by freight forwarders is the FCP. In this case, requests are not fulfilled on their own, but outsourced to common carriers. For the services forwarders buy from the common carriers freight charges have to be paid. The amount of these freight charges is oriented to the performance of the

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1_11,

services rather than to the actual costs of the common carriers. In other words, the freight charges are determined based on the characters of the requests. More precisely, the freight charges are calculated based on both the distance and the quantity related to a request. Due to the economies of scale, the cost rate decreases with increasing utilization of the transport resources and it makes sense for the forwarders to bundle the LTL requests before releasing them to the common carriers so that they will be billed less for the same requests. The FCP is modeled as MIP and two heuristic approaches to solve this problem are described. The effect of reducing freight charges through consolidation is significant in such situations, where the requests share the same pickup and delivery regions and the same direction. Computational results confirm the meaning of the FCP in such situations as well as the approaches' capability of finding high-quality solutions.

The next topic discussed in this thesis concerns the problem that arises when forwarders also use the transportation capacities of their subcontractors to fulfill their requests. The usage of external resources that are generally more expensive than the own vehicles enables forwarder to downsizing their fleet and enjoy more flexibility of available capacity. Due to the high fixed costs associated with vehicles, this strategy can also lead to considerable cost-savings over a long time horizon. However, the introduction of external capacities into the operational transportation planning resulted in a more complicated optimization problem than the basic decision problems discussed in Chapters 2 and 3. A widely used planning strategy in practice is to do "cherry-picking", which can lead to a high efficiency of the own vehicles but also high costs for outsourcing. In contrast, an integrated planning approach that considers all fulfillment modes can help forwarders find the cost-minimal plans. This problem that is denoted as IOTPP is formally defined in Chapter 4. An ALNS heuristic and an iterative approach are then proposed in Chapter 5 to solve the IOTPP. Computational results show that the complicated iterative approach performs better than the ALNS heuristic in terms of solution quality, which is a further development of the ALNS heuristic of Ropke and Pisinger (2006) that has proved very successful in solving the PDPTW, especially for larger instances as the CIOTP instances tested in Chapter 8.

The economies of scale in transportation logistics motivate forwarders to cooperate horizontally by pooling their requests and vehicles through which additional cost-saving potential due to the complementarity of requests from different coalition partners can be exploited. The study of the planning process in such horizontal coalitions of independent freight forwarders in this thesis focuses on the development of decentralized planning approaches that can deal with distributed information and decision-making competences on the one hand and can realize as much potential as possible on the other hand. This problem is introduced and formally defined in Chapter 6. A *route-based request exchange mechanism* proposed in Wang and Kopfer (2014) to solve this problem is described in detail in Chapter 7. This approach follows the Dantzig-Wolfe decomposition principles (Dantzig and Wolfe, 1960) and can preserve the autonomy of coalition members. The CTP results of the computational study show a very high efficiency of the approach in realizing the embedded cost-saving potential that can be identified by solving the routing problem centrally.

Based on the discussions on the two topics, i.e., IOTP and CTP, the transportation planning problem of forwarders in both vertical and horizontal cooperation is studied in Chapter 8. The route-based request exchange mechanism is extended by heterogeneous vehicle fleet of the forwarders which is composed of the own vehicles and the vehicles hired from subcontractors on both a tour basis and a daily basis. Computational results based on some theoretical instances show that the potential of cost-savings by introducing request exchange to the IOTP accounts to over 10% on average. The proposed decentralized planning approach can averagely realize over 90% of the cost-saving potential, while privacy and autonomy of coalition members remain protected.

Compared with static CTP, the research on dynamic CTP still stands in the very beginning phase. Chapters 9 and 10 are thus dedicated to an exploratory discussion about this topic. The purpose is to have an insight into this research filed and to appeal to more intensive studies on more related topics. For this purpose, the DCFPDPF is introduced in Chapter 9. Although this problem seems to be simple in static scenarios, it can be used as a good start in the research on the dynamic CTP.

In order to solve the DCFPDPF, two rolling horizon planning approaches are proposed in Chapter 9. The first one works with a predefined fixed interval between two planning processes. In the second approach, a new planning is triggered so long as a request becomes urgent according to predefined criterion. Both variants of the rolling horizon planning have been successfully used to solve the dynamic routing problems in literature.

A comprehensive computational study on the DCFPDPF is conduced in Chapter 10. The results of this study offer some interesting insights into CTP in dynamic environment. The benefits of CTP in dynamic situations have been verified in all tests with all tested configurations. According to different situations, CTP can realize up to about 8% cost-savings against planning without request exchange. Especially in difficult situations, for instance, the instances have a strong dynamism or the price level of the common carriers for subcontracting is high, CTP can help forwarders reduce more costs than in easier situation. It is further suggested that forwarders should try to use request information released by customer in advance to improve the planning results. Individual planning strategies that have proved successful can be used to derive good settings for the CTP in dynamic environments.

11.2 Outlook of future research

As already pointed out at the beginning of this thesis, the research conducted here is dedicated to gain an insight into the operational transportation planning of modern freight forwarding companies taking all possible options of request fulfillment into account and to appeal to more intensive studies in this research area. In this section, some interesting research topics are outlined.

In the IOTPP studied in this thesis, the vehicles of different fulfillment modes are heterogeneous due to their cost functions instead of other aspects. In future research, a further extension that should be studied is to consider a higher level of heterogeneity of the fleet additionally due to different equipment and thus the compatibility with individual requests, even when they are of the same fulfillment mode. Another interesting extension is to introduce different requests that can be executed only by specific vehicle modes. A practical example of this extension is that not all requests are allowed to be subcontracted, at least not to the common carriers. The reason can be either the requirement of the customers or that the forwarders do not want to subcontract the requests which are from their important customers.

Some general topics still have to be investigated in future research on the CTP in both static and dynamic situations. As reported in the computational study in Section 7.2.1, by combining different classes of PDPTW instances, the cost-saving potential of the resulting CTP instances differs strongly from the other. For future research on the strategic level it is a challenging task to investigate which situations are appropriate for CTP and what preconditions have to be fulfilled for a successful operation of the coalitions. Another task is to consider the strategic behavior of participants in horizontal coalitions. It has to be ensured that even when individual participants cheat, e.g. by reporting false evaluation values, the mechanism can still achieve the desired results. In order to provide this, it is important to develop an appropriate profit sharing scheme, especially for the CTP considered in this thesis with only limited information available.

The consideration of transshipment in the context of CTP should also be studied in the future. The most challenging task to be mastered is to synchronize the scheduling of vehicle routes which are generated by coalition members in a decentralized way. Transshipment points can be existing warehouses where the goods to be transferred can be stored shortly before they are picked up by another vehicle. This precedence restriction must be introduced into the route generation process of the CTP. If the goods can be transferred between two vehicles at any place without extra facilities like cranes, vehicles have to be additionally scheduled in such a way that they must be at the same place at the same time. As a result, synchronizing vehicles in this situation will be more difficult.

Compared to the static CTP, more work must be done in the future in the study on the dynamic CTP. For the DCFPDPF studied in Chapters 9 and 10, an interesting task of the future research is to study how to specify good configurations for the rolling horizon planning approaches based on different characters of instances. Another important factor is the weight function that deals with the evaluation of the information of requests offered by customers in advance. The evaluation scheme used in the conducted computational study here, especially the setting of FLP-II, has proved to be appropriate. However, in the future research, it is interesting to test more weight functions, particularly when the requests have significant different characters as in this study, including travel distances of the requests, width of time windows, as well as distribution of the lead time t_r^{ld}.

A limitation of the proposed approaches in Chapter 9 for the DCFPDPF is that they focus only on the latest possible time to start the service of each request. The earliest time, which is the beginning of the time window is ignored in these approaches. In the future research, the possibility of assigning requests with wide time windows that can be fulfilled before they become due requests has to be considered in extensions of the proposed rolling horizon planning approaches.

It will be interesting and quite challenging to consider the LTL requests in the future research on the dynamic CTP. Also the integration of the two fulfillment modes related to subcontractors offering vehicles on both a tour and a daily basis may be an interesting topic in this research field.

In the research on the IOTPP and the CTP in Chapters 4 to 10, a simplification is made concerning the fulfillment mode subcontracting on common carriers. It is assumed that all requests are outsourced singly without being consolidated. In the future research, the FCP can also be considered as a component of the optimization problems in these topics. Specifically, in the CTP, the problem can be solved by the agent, who can generated efficient request bundles to be outsourced to common carrier while the forwarders in the coalition still only have to concentrate on their vehicles.

Bibliography

Applegate D. L., Bixby R. E., Chvátal V., Cook W. J. (2006) The traveling salesman problem - A computational study. Princeton University Press, Princeton Oxford

Baldacci R., Battarra M., Vigo D. (2008) Routing a heterogeneous fleet of vehicles. In: Golden B. L., Raghaven S., Wasil E. A. (eds) The Vehicle Routing Problem - Latest Advances and New Challenges, Operations Research/Computer Science Interfaces, vol 43, Springer US, pp. 3–27

Baldacci R., Bertolini E., Mingozzi A. (2011) An exact algorithm for the pickup and delivery problem with time windows. Operations Research 59:414–426

Ball M. O., Golden B. L., Assad A. A., Bodin L. D. (1983) Planning for truck fleet size in the presence of a common-carrier option. Decision Science 14:103–120

Ballou R. H. (1999) Bsuness Logistics Management, fourth international ed. Prentice-Hall International, Inc., New Jersey

Bent R., van Hentenryck P. (2003) A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. In: Rossi F. (ed) Principles and Practice of Constraint Programming – CP 2003, Springer, Berlin Heidelberg, Lecture Notes in Computer Science, vol 2833, pp. 123–137

Bent R., van Hentenryck P. (2006) A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. Computers & Operations Research 33:875–893

Berbeglia G., Cordeau J.-F., Gribkovskaia I., Laporte G. (2007) Static pickup and delivery problems: a classification scheme and survey. Top 15:1–31

Berbeglia G., Cordeau J. F., Laporte G. (2010) Dynamic pickup and delivery problems. European Journal of Operational Research 202:8–15

Berger S., Bierwirth C. (2010) Solutions to the request reassignment in collaborative carrier networks. Transportation Research Part E 46:627–638

Bolduc M.-C., Renaud J., Boctor F. (2007) A heuristic for the routing and carrier selection problem. European Journal of Operational Research 183:926–932

X. Wang, *Operational Transportation Planning of Modern Freight Forwarding Companies,* Produktion und Logistik, DOI 10.1007/978-3-658-06869-1,

Bolduc M.-C., Renaud J., Boctor F., Laporte G. (2008) A perturbation metaheuristic for the vehicle routing problem with private fleet and common carriers. Journal of the Operational Research Society 59:776–787

Bramel J., Simchi-Levi D. (2002) Set-covering-based algorithms for the capacitated VRP. In: Toth P., Vigo D. (eds) The Vehicle Routing Problem, SIAM Monographs on Discrete Mathematics and Applications, vol 9, SIAM, Philadelphia, PA, chap 4, pp. 85–108

Brandão J. (2009) A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. European Journal of Operational Research 195:716–728

Butt S. E., Ryan D. M. (1999) An optimal solution procedure for the multiple tour maximum collection problem using column generation. Computers & Operations Research 26:427–441

Ceschia S., Gaspero L. D., Schaerf A. (2011) Tabu search techniques for the heterogeneous vehicle routing problem with time windows and carrier-dependent costs. Journal of Scheduling 14:601–615

Chen Z.-L., Xu H. (2006) Dynamic column generation for dynamic vehicle routing with time windows. Transportation Science 40:74–88

Chu C.-W. (2005) A heuristic algorithm for the truckload and less-than-truckload problem. European Journal of Operational Research 165:657–667

Clifton C., Iyer A., Cho R., Jiang W., Kantarcıoğlu M., Vaidya J. (2008) An approach to securely identifying beneficial collaboration in decentralized logistics systems. Manufactureing & Service Operations Management 10:108–125

Cordeau J.-F. (2006) A branch-and-cut algorithm for the dial-a-ride problem. Operations Research 54:573–586

Cordeau J.-F., Laporte G., Savelsbergh M. W., Vigo D. (2007) Vehicle routing. In: Barnhart C., Laporte G. (eds) Transportation, Handbooks in Operations Research and Management Science, vol 14, Elsevier B.V., Amsterdam, chap 6, pp. 367–428

Cordeau J.-F., Laporte G., Ropke S. (2008) Recent models and algorithms for one-to-one pickup and delivery problems. In: Golden B. L., Raghaven S., Wasil E. A. (eds) The Vehicle Routing Problem - Latest Advances and New Challenges, Operations Research/Computer Science Interfaces, vol 43, Springer US, pp. 327–357

Côté J.-F., Potvin J.-Y. (2009) A tabu search heuristic for the vehicle routing problem with private fleet and common carrier. European Journal of Operational Research 198:464–469

Cruijssen F., Salomon M. (2004) Empirical study: Order sharing between transportation companies may result in cost reductions between 5 to 15 percent. In: CentER Discussion Paper No. 2004-80, Tilburg University

Cruijssen F., Bräysy O., Dullaert W., Fleuren H., Salomon M. (2007a) Joint route planning under varying market conditions. International Journal of Physical Distribution & Logistics Management 37:287–304

Cruijssen F., Cools M., Dullaert W. (2007b) Horizontal cooperation in logistics: Opportunities and impediments. Transportation Research Part E 43:129–142

Cruijssen F., Dullaert W., Fleuren H. (2007c) Horizontal cooperation in transport and logistics: a literature review. Transportation Journal 46:22–39

Dantzig G. B., Ramser J. H. (1959) The truck dispatching problem. Management Science 6:80–91

Dantzig G. B., Wolfe P. (1960) Decomposition principle for linear programs. Operations Research 8:101–111

Desaulniers G., Desrosiers J., Erdmann A., Solomon M. M., Soumis F. (2002) VRP with pickup and delivery. In: Toth P., Vigo D. (eds) The Vehicle Routing Problem, SIAM, Philadelphia, PA, SIAM Monographs on Discrete Mathematics and Applications, vol 9, pp. 225–242

Dowsland K. A. (1995) Simulated annealing. In: Reeves C. (ed) Modern heuristic techniques for combinatorial problems, McGraw-Hill, London, pp. 20–69

Dumas Y., Desrosiers J., Soumis F. (1991) The pickup and delivery problem with time windows. European Journal of Operational Research 54:7–22

Feillet D., Dejax P., Gendreau M. (2005) Traveling salesman problem with profits. Transportation Science 39:188–205

Gendreau M., Guertin F., Potvin J.-Y., Taillard E. (1999) Parallel tabu search for real-time vehicle routing and dispatching. Transportation Science 33:381–390

Gendreau M., Guertin F., Potvin J.-Y., Séguin R. (2006) Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries. Transportation Research Part C 14:157–174

Glover F. (1996) Ejection chains, reference structures and alternating path methods for traveling salesman problems. Discrete Applied Mathematics 65:223–253

Golden B. L., Raghaven S., Wasil E. A. (eds) (2008) The Vehicle Routing Problem: Latest Advances and New Challenges. Springer, New York

Gutin G., Punnen A. P. (eds) (2002) The traveling salesman problem and its variations. Kluwer Academic Publishers, Dordrecht

Hoff A., Andersson H., Christiansen M., Hasler G., Løkketangen A. (2010) Industrial aspects and literature survey: Fleet composition and routing. Computers & Operations Research 37:2041–2061

Klincewicz J. G., Luss H., Pilcher M. G. (1990) Fleet size planning when outside carrier services are available. Transportation Science 24:169–182

Kopfer H. (1990) Der Entwurf und die Realisierung eines A*-Verfahrens zur Lösung des Frachtoptimierungsproblem. OR Spektrum 12:207–218

Kopfer H. (1992) Konzepte genetischer algorithmen und ihre anwendung auf das frachtoptimierungsproblem im gewerblichen Güterverkerh. OR Spektrum 14:137–147

Kopfer H., Pankratz G. (1999) Das Groupage-Problem kooperierender Verkehrs- träger. In: Kall P., Lüthi H.-J. (eds) Operations Research Proceedings 1998, Springer, Berlin Heidelberg New York, pp. 453–462

Kopfer H., Wang X. (2009) Combining vehicle routing with forwarding - extension of the vehicle routing problem by different types of subcontraction. Journal of the Korean Institute of Industrial Engineers 35:1–14

Kopfer H., Krajewska M. A., Jurczyk A. (2006) Kooperation von Profit Centern in Transportwesen: Endbericht des Forschungsprojektes “Integrierte Auftragsdisposition in eigenständigen Profit Centern eines Speditionsunternehmens (INAPCE)”. Tech. rep., Bremer Innovationsagentur, Bremen

Krajewska M., Kopfer H. (2006) Collaborating freight forwarding enterprises - request allocation and profit sharing. OR Spectrum 28:301–317

Krajewska M., Kopfer H. (2009) Transportation planning in freight forwarding companies - tabu search algorithm for the integrated operational transportation planning problem. European Journal of Operational Research 197:741–751

Krajewska M. A. (2008) Potentials for efficiency increase in modern freight forwarding. Betribswirtschaftlicher Verlag Dr. Th. Gabler, Wiesbaden

Krajewska M. A., Kopfer H., Laporte G., Ropke S., Zaccour G. (2008) Horizontal cooperation of freight carriers: request allocation and profit sharing. Journal of the Operational Research Society 59:1483–1491

Laporte G. (1992) The traveling salesman problem: An overview of exact and approximate algorithms. European Journal of Operational Research 59:231–247

Laporte G. (2009) Fifty years of vehicle routing. Transportation Science 43:408–416

Larsen A. (2001) The dynamic vehicle routing problem. PhD thesis, Technical University of Denmark (DTU)

Lau H. C., Liang Z. (2002) Pickup and delivery with time windows: Algorithms and test case generation. International Journal on Artificial Intelligent Tools 11:455–472

Lawler E. L., Shmoys D. B., Rinnooy Kan A. H. G., Lenstra J. K. (eds) (1985) The traveling salesman problem. Series in discrete mathematics and optimization, Wiley-Interscience, Chichester New York Brisbane Toronto Singapore

Lee C.-G., Kwon R. H., Ma Z. (2007) A carrier's optimal bid generation problem in combinatorial auctions for transportation procurement. Transportation Research Part E 43:173–191

Li F., Golden B., Wasil E. (2007) A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem. Computers & Operations Research 34:2734–2742

Li H., Lim A. (2001) A metaheuristic for the pickup and delivery problem with time windows. In: ICTAI-2001, The 13th IEEE conference on Tools with artificial intelligence, Dallas, TX, pp. 160–170

Lu Q., Dessouky M. (2004) An exact algorithm for the multiple vehicle pickup and delivery problem. Transportation Science 38:503–514

Lu Q., Dessouky M. M. (2006) A new insertion-based construction heuristic for solving the pickup and delivery problem with time windows. European Journal of Operational Research 175:672–687

Lund K., Madsen O. B. G., Rygaard J. M. (1996) Vehicle routing problems with varying degrees of dynamism. Tech. rep., IMM Institute of Mathematical Modelling

Masson R., Lehuédé F., Péton O. (2012) The dial-a-ride problem with transfers. Tech. rep., Lunam Université, École des Mines de Nantes, IRCCyN

Masson R., Lehuédé F., Péton O. (2013) An adaptive large neighborhood search for the pickup and delivery problem with transfers. Transportation Science 47:344–355

Mitrović-Minić S., Laporte G. (2004) Waiting strategies for the dynamic pickup and delivery problem with time windows. Transportation Research Part B 38:635–655

Montemanni R., Gambardella L. M., Rizzoli A. E., Donati A. V. (2005) Ant colony system for a dynamic vehicle routing problem. Journal of Combinatorial Optimization 10:327–343

Mues C., Pickl S. (2002) Pickup & Delivery-Probleme mit Umlademöglichkeit Ein Tourenplanungsproblem aus der Automobilzulieferindustrie. In: Chamoni P., Leisten R., Martin A., Minnemann J., Stadtler H. (eds) Operations Research Proceedings 2011, Springer Berlin Heidelberg, pp. 33–40

Mues C., Pickl S. (2005) Transshipment and time windows in vehicle routing. In: Bein W., Chin F., Hsu D., Palis M. (eds) ISPAN'05: Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks, pp. 113–119

Müller-Merbach H. (1983) Zweimal travelling salesman. DGOR-Bulletin 25:12–13

Nanry W. P., Barnes J. W. (2000) Solving the pickup and delivery problem with time windows using reactive tabu search. Transportation Research Part B 34:107–121

Özener O. O., Ergun O., Savelsbergh M. (2011) Lane-exchange mechanisms for truckload carrier collaboration. Transportation Science 45:1–17

Pankratz G. (2005) A grouping genetic algorithm for the pickup and delivery problem with time windows. OR Spectrum 27:21–41

Parragh S. N., Doerner K. F., Hartl R. F. (2008a) A survey on pickup and delivery problems - Part I: Transportation bwteen customers and depot. Journal für Betriebswirtschaft 58:21–51

Parragh S. N., Doerner K. F., Hartl R. F. (2008b) A survey on pickup and delivery problems - Part II: Transportation between pickup and delivery locations. Journal für Betriebswirtschaft 58:81–117

Petersen H. L., Ropke S. (2011) The pickup and delivery problem with cross-docking opportunity. In: Böse J., Hu H., Jahn C., Shi X., Stahlbock R., Voß S. (eds) Computational Logistics, Springer-Verlag Berlin Heidelberg, LNCS, vol 6971, pp. 101–113

Pillac V., Gendreau M., Guéret C., Medaglia A. L. (2013) A review of dynamic vehicle routing problems. European Journal of Operational Research 225:1–11

Psaraftis H. N. (1980) A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. Transportation Science 14:130–154

Renaud J., Boctor F. F., Ouenniche J. (2000) A heuristic for the pickup and delivery traveling salesman problem. Computers & Operations Research 27:905–916

Ropke S., Cordeau J.-F. (2009) Branch and cut and price for the pickup and delivery problem with time windows. Transportation Science 43:267–286

Ropke S., Pisinger D. (2006) An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science 40:445–472

Ropke S., Cordeau J.-F., Laporte G. (2007) Models and branch-and-cut algorithms for pickup and delivery problems with time windows. Networks 49:258–272

Savelsbergh M., Sol M. (1998) DRIVE: Dynamic Routing of Independent VEhicles. Operations Research 46:474–490

Savelsbergh M. W. P., Sol M. (1995) The general pickup and delivery problem. Transportation Science 29:17–29

Schönberger J. (2005) Operational Freight Carrier Planning. Springer, Berlin Heidelberg New York

Schrijver A. (2005) On the history of combinatorial optimization (till 1960). In: K. Aardal G. N., Weismantel R. (eds) Discrete Optimization, Handbooks in Operations Research and Management Science, vol 12, Elsevier, pp. 1 – 68

Schwind M., Gujo O., Vykoukal J. (2009) A combinatorial intra-enterprise exchange for logistics services. Information Systems and E-Business Management 7:447–471

Shaw P. (1997) A new local search algorithm providing high quality solutions to vehicle routing problems. Tech. rep., Department of Computer Science, University of Strathclyde, Scotland

Shaw P. (1998) Using constraint programming and local search methods to solve vehicle routing problems. In: Maher M., Puget J.-F. (eds) Principles and Practice of Constraint Programming - CP98, Springer, Berlin Heidelberg, LNCS, vol 1520, pp. 417–431

Sigurd M., Pisinger D., Sig M. (2004) Scheduling transportation of live animals to avoid the spread of diseases. Transportation Science 38:197–209

Solomon M. M. (1987) Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research 35:254–265

Song J., Regan A. (2003) An auction based collaborative carrier network. Tech. Rep. UCI-ITS-WP-03-6, Institute of Transportation Studies, University of California, Irvine

Song J., Regan A. (2005) Approximation algorithms for the bid construction problem in combinatorial auctions for the procurement of freight transportation contracts. Transportation Research Part E 39:914–933

Stadtler H. (2009) A framework for collaborative planning and state-of-the-art. OR Spectrum 31:5–30

Strumpf P. (1998) Tourenplanung im speditionellen Güterverkehr. Gesellschaft für Verkehrsbetriebswirtschaft und Logistik (GVB) e.V., Nürnberg

Struwe E., Wang X., Sander C., Kopfer H. (2012) A new simulated annealing algorithm for the freight consolidation problem. In: Klatte D., Lüthi H.-J., Schmedders K. (eds) Operations Research Proceedings 2011, Springer, Berlin Heidelberg, Operations Research Proceedings, pp. 299–304

Taillard E., Badeau P., Gendreau M., Guertin F., Potvin J.-Y. (1997) A tabu search heuristi for the vehicle routing problem with soft time windows. Transportation Science 31:170–186

Tjokroamidjojo D., Kutanoglu E., Taylor G. D. (2006) Quantifying the value of advance load information in truckload trucking. Transportation Research Part E 42:340–357

Toth P., Vigo D. (eds) (2002a) The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications, SIAM, Philadelphia

Toth P., Vigo D. (2002b) VRP with backhauls. In: Toth P., Vigo D. (eds) The Vehicle Routing Problem, SIAM Monographs on Discrete Mathematics and Applications, vol 9, SIAM, Philadelphia, PA, chap 8, pp. 195–224

Vickrey W. (1961) Counterspeculation, auctions, and competitive sealed tenders. Journal of Finance 16:8–37

Wang X., Kopfer H. (2011) Increasing efficiency of freight carriers through collaborative transport planning: Chances and challenges. In: Ivanov D., Kopfer H., Haasis H.-D., Schönberger J. (eds) Dynamics and Sustainability in International Logistics and Supply Chain Management - Proceedings of the 6th German-Russian Logistics and SCM Workshop (DR-LOG 2011), Cuvillier Verlag, Göttingen, pp. 41–50

Wang X., Kopfer H. (2013a) Dynamic collaborative transportation planning: A rolling horizon planning approach. In: Pacino D., Voss S., Jensen R. M. (eds) Proceedings of ICCL 2013, LNCS, vol 8197, Springer, Berlin Heidelberg, pp. 128–142

Wang X., Kopfer H. (2013b) Rolling horizon planning approaches for a dynamic collaborative routing problem with full truckload pickup and delivery requests. Tech. rep., Chair of Logistics Working Paper No. 6, University of Bremen, Bremen

Wang X., Kopfer H. (2014) Collaborative transportation planning of less-than-truckload freight: A route-based request exchange mechanism. OR Spectrum 36:357–380

Wang X., Kopfer H., Gendreau M. (2014) Operational transportation planning of freight forwarding companies in horizontal coalitions. European Journal of Operational Research 237:1133–1141

Xu H., Chen Z.-L., Rajagopal S., Arunapuram S. (2003) Solving a practial pickup and delivery problem. Transportation Science 37:347–364

Yang J., Jaillet P., Mahmassani H. S. (1998) On-line algorithms for truck fleet assignment and scheduling under real-time information. Transportation Research Record 1667:107–113

Yang J., Jaillet P., Mahmassani H. S. (2004) Real-time multivehicle truckload pickup and delivery problems. Transportation Science 38:135–148