Chapter 2: Linear Signal Models

Motivation

- Signal modeling is the representation of signal in an efficient manner.
- Signal modeling has several applications.
  - Compression,
    - Represent the signal with small number of parameters.
  - Signal prediction,
    - A model may be used to estimate future data.
  - Signal interpolation,
    - A model may be used to estimate missing data.

- **Nonparametric signal model:** when the LTI filter is specified by its impulse response.
  - Non parametric because there is no restriction regarding the form of the model and the number of parameters is infinite.
- **Parametric:** when the filter is represented by a finite-order rational system function.
  - Limited to
    - System with rational system function: All-pole, All-zero and pole-zero systems.
    - Minimum-phase systems
  - Described by a finite number of parameters.

- The two major topics we address in this chapter are
  - Design of an all-pole, all-zero, or pole-zero system that produces a random signal with a given autocorrelation sequence or PSD function.
  - Derivation of the second-order moments, given the parameters of their system function, and

Nonparametric Signal Models

- The model given below is a nonparametric model.

$$x(n) = \sum_{k=-\infty}^{\infty} h(k)w(n-k)$$

- If w(n) is a zero-mean white noise process the autocorrelation, complex PSD and PSD of the output for **unit sample input** are given as

$$r_x(l) = \sigma_w^2 \sum_{k=-\infty}^{\infty} h(k)h^*(k-l) = \sigma_w^2 r_h(l)$$

$$R_x(z) = \sigma_w^2 \, H(z)H^*\left(\frac{1}{z^*}\right)$$

$$R_x(e^{j\omega}) = \sigma_w^2 \, |H(e^{j\omega})|^2 = \sigma_w^2 \, R_h(e^{j\omega})$$

- Notice that when the input is a white noise process, the shape of the autocorrelation and the power spectrum (*second-order moments*) of the output signal are completely characterized by the system.
- We use the term *system-based signal model* to refer to the signal generated by a system with a white noise input.

- It can also be represented recursively with

$$x(n) = -\sum_{k=1}^{\infty} h_I(k)x(n-k) + w(n)$$

- Where $h_I(m)$ is the inverse system of h(n).

$$X(z) = H(z)W(z)$$

$$X(z)HI(z) = W(z)$$

$$\sum_0^{\infty} x(n-k)h_I(k) = w(n)$$

- Assuming $h_I(0)=1$ leads to the equation.

- Another way to look at it is with the following equation.

$$x(n+1) = \underbrace{\sum_{k=-\infty}^{n} h(n+1-k)x(k)}_{\text{past information: linear combination of } x(n),\ x(n-1),\ldots} + \underbrace{w(n+1)}_{\text{new information}}$$

Parametric Signal Modeling

- Two steps in parametric signal modeling:
  - Choose an appropriate model and
  - Determine the parameters of the model.
- In linear signal model, the signal is the output of a causal stable LTI filter.

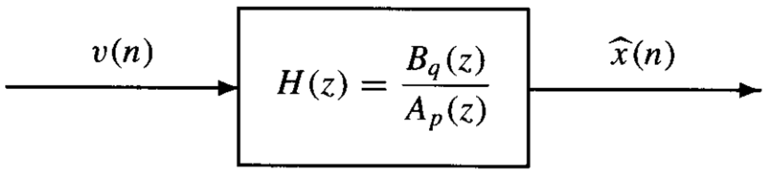
$$H(z) = \frac{B_q(z)}{A_p(z)} = \frac{\sum_{k=0}^{q} b_q(k) z^{-k}}{1 + \sum_{k=1}^{p} a_p(k) z^{-k}}$$

- Parameters are $a_p(k)$ and $b_q(k)$.

- The parameters should provide the best approximation to the given signal.
- What does causality and stability imply in this system function?

Deterministic Signal Modeling

- The signal is modeled as an output to an LTI system with impulse input.



- What do we mean by best approximation?
  - Error or Implementation cost
- The error in this modeling is given as

$$e'(n) = x(n) - \hat{x}(n)$$

- Different methods of minimizing this error.

Least Square Method

- It tries to minimize the square of the error.

$$\mathcal{E}_{LS} = \sum_{n=0}^{\infty} \left| e'(n) \right|^2$$

- By taking the partial derivative of this error with respect to $a_p(k)$ and $b_q(k)$ and setting to zero leads to nonlinear equations that are not mathematically tractable.

$$\frac{\partial \mathcal{E}_{LS}}{\partial a_p^*(k)} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left[X(e^{j\omega}) - \frac{B_q(e^{j\omega})}{A_p(e^{j\omega})} \right] \frac{B_q^*(e^{j\omega})}{[A_p^*(e^{j\omega})]^2} e^{jk\omega} d\omega = 0$$

$$\frac{\partial \mathcal{E}_{LS}}{\partial b_q^*(k)} = -\frac{1}{2\pi} \int_{-\pi}^{\pi} \left[X(e^{j\omega}) - \frac{B_q(e^{j\omega})}{A_p(e^{j\omega})} \right] \frac{e^{jk\omega}}{A_p^*(e^{j\omega})} d\omega = 0$$

Padé Approximation

- Reformulates the problem in such a way that the filter coefficients may be found that force the **error to be zero** for p+q+1 values of n.

$$x(n) + \sum_{k=1}^{p} a_p(k)x(n-k) = \begin{cases} b_q(n) & ; \quad n = 0, 1, \ldots, q \\ 0 & ; \quad n = q+1, \ldots, q+p \end{cases}$$

- In matrix form,

Denominator coefficients

$$\begin{bmatrix} x(q) & x(q-1) & \cdots & x(q-p+1) \\ x(q+1) & x(q) & \cdots & x(q-p+2) \\ \vdots & \vdots & & \vdots \\ x(q+p-1) & x(q+p-2) & \cdots & x(q) \end{bmatrix} \begin{bmatrix} a_p(1) \\ a_p(2) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} x(q+1) \\ x(q+2) \\ \vdots \\ x(q+p) \end{bmatrix}$$

Numerator coefficients

$$\begin{bmatrix} x(0) & 0 & 0 & \cdots & 0 \\ x(1) & x(0) & 0 & \cdots & 0 \\ x(2) & x(1) & x(0) & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ x(q) & x(q-1) & x(q-2) & \cdots & x(q-p) \end{bmatrix} \begin{bmatrix} 1 \\ a_p(1) \\ a_p(2) \\ \vdots \\ a_p(p) \end{bmatrix} = \begin{bmatrix} b_q(0) \\ b_q(1) \\ b_q(2) \\ \vdots \\ b_q(q) \end{bmatrix}$$

- Solution different depending upon whether the matrix is invertible or not.
  - Reading assignment.
- Limitation
  - No guarantee on how accurate the model is for $n>p+q$ since no data for $n>p+q$ is used.
  - The model generated is not guaranteed to be stable.

Prony's Method

- Modifies the least square error definition,

$$E(z) = \boxed{A_p(z)E'(z)} = A_p(z)X(z) - B_q(z)$$

- The new error is then given as

$$e(n) = a_p(n) * x(n) - b_q(n) = \hat{b}_q(n) - b_q(n)$$

$$e(n) = \begin{cases} x(n) + \sum_{l=1}^{p} a_p(l)x(n-l) - b_q(n) & ; \quad n = 0, 1, \ldots, q \\ x(n) + \sum_{l=1}^{p} a_p(l)x(n-l) & ; \quad n > q \end{cases}$$

- Then minimizing the square of this error with respect to $a_p(k)$.

- By setting the partial derivative to zero.

$$\sum_{n=q+1}^{\infty} e(n)x^*(n-k) = 0 \quad ; \quad k = 1, 2, \ldots, p$$

Orthogonally Principle

- Substituting this into the modified error equation and manipulation will lead to:

$$\sum_{l=1}^{p} a_p(l) r_x(k,l) = -r_x(k,0) \quad ; \quad k = 1, 2, \ldots, p$$

- Where

$$r_x(k,l) = \sum_{n=q+1}^{\infty} x(n-l)x^*(n-k)$$

- In matrix form,

$$\begin{bmatrix} r_x(1,1) & r_x(1,2) & r_x(1,3) & \cdots & r_x(1,p) \\ r_x(2,1) & r_x(2,2) & r_x(2,3) & \cdots & r_x(2,p) \\ r_x(3,1) & r_x(3,2) & r_x(3,3) & \cdots & r_x(3,p) \\ \vdots & \vdots & \vdots & & \vdots \\ r_x(p,1) & r_x(p,2) & r_x(p,3) & \cdots & r_x(p,p) \end{bmatrix} \begin{bmatrix} a_p(1) \\ a_p(2) \\ a_p(3) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} r_x(1,0) \\ r_x(2,0) \\ r_x(3,0) \\ \vdots \\ r_x(p,0) \end{bmatrix}$$

↑

Conjugate symmetric matrix

- The minimum error is then

$$\epsilon_{p,q} = r_x(0,0) + \sum_{k=1}^{p} a_p(k) r_x(0,k)$$

- minimum error is not dependent on the numerator coefficients

- The $b_q(k)$ are obtained by setting the error to zero for n=0,…,q. Same as Padé approx.

$$b_q(n) = x(n) + \sum_{k=1}^{p} a_p(k)x(n-k)$$

- Limitation
  - Requires knowledge of data for all n.

Padé vs Prony

- Model a signal consisting of single pulse of length N=21 with p=q=1.

$$x(n) = \begin{cases} 1 & ; \; n = 0, 1, \ldots, N-1 \\ 0 & ; \; \text{else} \end{cases}$$

- Prony's method

$$H(z) = \frac{1 + 0.05z^{-1}}{1 - 0.95z^{-1}}$$

$$h(n) = \delta(n) + (0.95)^{n-1} u(n-1)$$

- Padé method

$$H(z) = \frac{1}{1 - z^{-1}}$$

$$h(n) = u(n)$$

Filter Design: Padé vs Prony

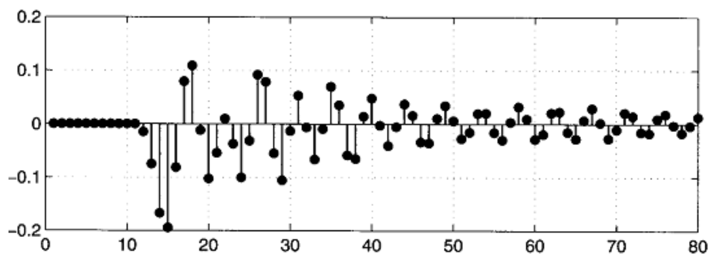
- Design a linear phase lowpass filter for

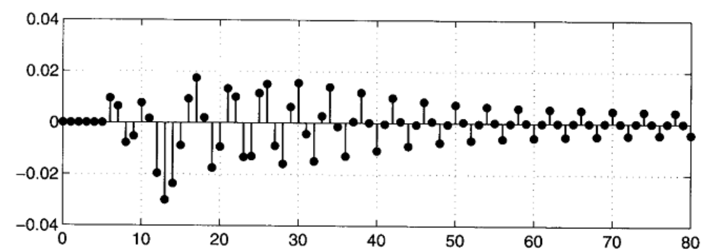
$$I(e^{j\omega}) = \begin{cases} e^{-jn_d\omega} & ; \quad |\omega| < \pi/2 \\ 0 & ; \quad \text{otherwise} \end{cases} \qquad i(n) = \frac{\sin\left[(n-n_d)\pi/2\right]}{(n-n_d)\pi}$$

Pade method

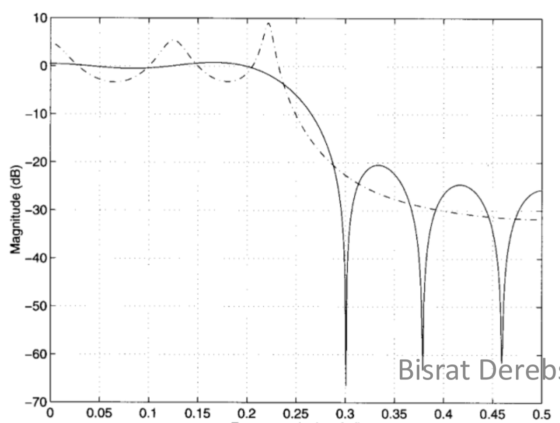
Prony method

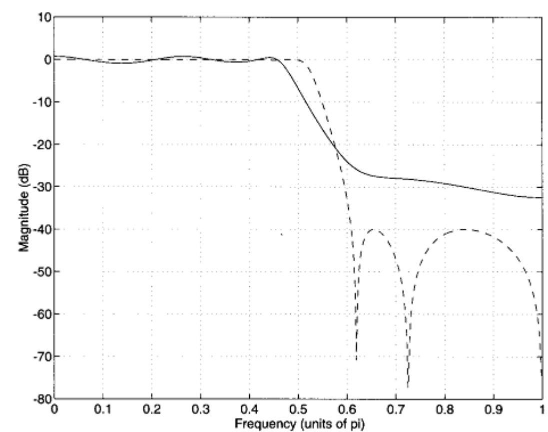
Error





Magnitude Response





Shanks' Method

- Instead of forcing the error zero to obtain $b_q(k)$, Shanks' method minimizes the squared error.

$$\mathcal{E}_S = \sum_{n=0}^{\infty} \left| e'(n) \right|^2$$

- Reading assignment.

Reading assignment

- FIR least square invers filter
  - Hayes: pp 166-174
- Iterative prefiltering
  - Hayes: pp 174-177

Finite Data Records

- If only x(n) is known for finite interval $0 \leq n \leq N$, Prony's method cannot be used.

$$\sum_{l=1}^{p} a_p(l) r_x(k,l) = -r_x(k,0) \quad ; \quad k = 1, 2, \ldots, p$$

$$r_x(k,l) = \sum_{n=q+1}^{\infty} x(n-l)x^*(n-k)$$

- Assumption have to be made regarding x(n) outside this interval.

Autocorrelation Method

- In this method, the data is assumed to be zero outside 0≤n≤N.
- This is basically using a rectangular window to obtain a new signal from x(n).

$$\tilde{x}(n) = x(n)w_R(n)$$

$$w_R(n) = \begin{cases} 1 & ; \quad n = 0, 1, \ldots, N \\ 0 & ; \quad \text{otherwise} \end{cases}$$

- Then solve for $a_p(k)$ using Prony's method except

$$\sum_{l=1}^{p} a_p(l) r_x(k-l) = -r_x(k) \quad ; \quad k = 1, 2, \ldots, p$$

$$r_x(k) = \sum_{n=k}^{N} x(n) x^*(n-k) \quad ; \quad k \geq 0$$

- Limitation
  - Since the window forces $x(n)$ to be zero outside $0 \le n \le N$, the accuracy of the model outside this range is compromised.
  - For $0 \le n \le P$ the prediction is based on fewer data, so the error may be greater.
    - A non-rectangular window may be used. Hamming, Hanning, …
- Advantage
  - The model will be stable. That is the poles of $H(z)$ will be inside the unit circle.

Covariance Method

- Does not make any assumptions about the data outside the 0≤n≤N.
- Error is calculated for data P≤n≤N.

$$\mathcal{E}_p^C = \sum_{n=p}^{N} |e(n)|^2$$

- The solution is then given as

$$\begin{bmatrix} r_x(1,1) & r_x(1,2) & r_x(1,3) & \cdots & r_x(1,p) \\ r_x(2,1) & r_x(2,2) & r_x(2,3) & \cdots & r_x(2,p) \\ r_x(3,1) & r_x(3,2) & r_x(3,3) & \cdots & r_x(3,p) \\ \vdots & \vdots & \vdots & & \vdots \\ r_x(p,1) & r_x(p,2) & r_x(p,3) & \cdots & r_x(p,p) \end{bmatrix} \begin{bmatrix} a_p(1) \\ a_p(2) \\ a_p(3) \\ \vdots \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} r_x(1,0) \\ r_x(2,0) \\ r_x(3,0) \\ \vdots \\ r_x(p,0) \end{bmatrix}$$

- Where $r_x(k,l) = \sum_{n=p}^{N} x(n-l)x^*(n-k)$

- The normal equations are identical to those of Prony's method, except how the autocorrelation is obtained.
- Note that the autocorrelation matrix is not Toeplitz.
  - Computationally much costly than autocorrelation method.

Example: Autocorr. vs Covariance

- Obtain a first order all-pole model for

$$\mathbf{x} = \left[1,\ \beta,\ \beta^2, \ldots, \beta^N\right]^T$$

$$H(z) = \frac{b(0)}{1 + a(1)z^{-1}}$$

- Autocorrelation

$$a(1) = -\beta \frac{1 - |\beta|^{2N}}{1 - |\beta|^{2N+2}}$$

$$\lim_{N \to \infty} a(1) = -\beta$$

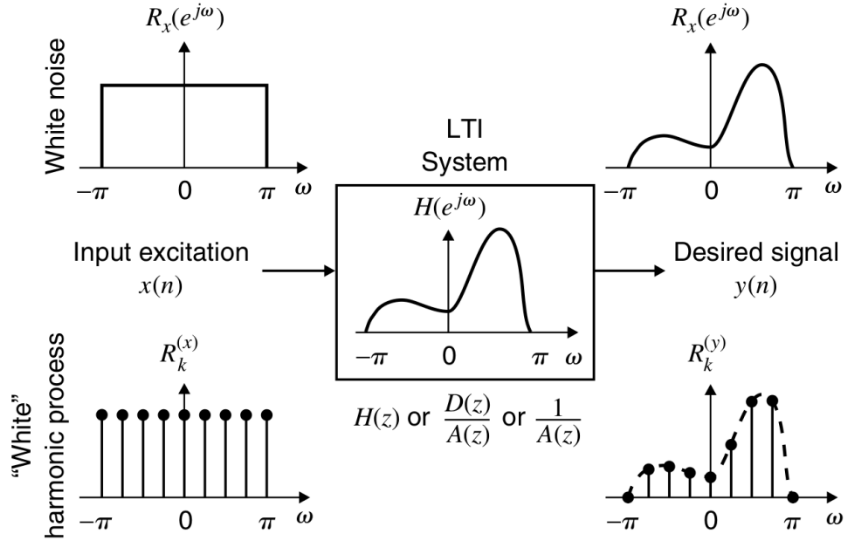
- Covariance

$$a(1) = -\beta$$

Stochastic Signal Modeling

- Analyze the properties of a special class of stationary random sequences that are obtained by driving a linear, time-invariant system with white noise.
- Previous methods not applicable for stochastic signal modeling.
  - Only probabilistic information is available.
  - Errors are also described probabilistically.

- If a white noise is filtered with a stable LTI filter, random signals with almost any arbitrary a periodic correlation structure or continuous PSD can be obtained.



Autoregressive Moving-Average (Pole-Zero) Models

- A pole-zero stochastic model is given as

$$x(n) + \sum_{k=1}^{P} a_k\, x(n-k) = \sum_{k=0}^{Q} d_k\, w(n-k)$$

- Its system function is:

$$H(z) = \frac{X(z)}{W(z)} = \frac{\sum_{k=0}^{Q} d_k z^{-k}}{1 + \sum_{k=1}^{P} a_k z^{-k}} \triangleq \frac{D(z)}{A(z)}$$

- Its short term memory is exponentially decaying

$$h(n) = \sum_{j=0}^{Q-P} B_j \delta(n-j) + \sum_{k=1}^{P_1} A_k (p_k)^n u(n) + \sum_{i=1}^{P_2} C_i r_i^n \cos(\omega_i n + \phi_i)\, u(n)$$

- If a parametric model is excited with white noise w(n)=IID{0, σ^2_ω}, the second moments of the random signal is given as

$$r_x(l) = \sigma_w^2 r_h(l) = \sigma_w^2 h(l) * h^*(-l)$$

$$R_x(z) = \sigma_w^2 R_h(z) = \sigma_w^2 H(z) H^* \left(\frac{1}{z^*}\right)$$

$$R_x(e^{j\omega}) = \sigma_w^2 R_h(e^{j\omega}) = \sigma_w^2 |H(e^{j\omega})|^2$$

All-pole Models

- An all pole model is when Q=0, that is

$$H(z) = \frac{d_0}{A(z)} = \frac{d_0}{1 + \sum_{k=1}^{P} a_k z^{-k}} = \frac{d_0}{\prod_{k=1}^{P} (1 - p_k z^{-1})}$$

- Its impulse response is given by

$$h(n) = -\sum_{k=1}^{P} a_k h(n-k) + d_0 \delta(n)$$

- Multiplying by *h*(n-l)* and summing for all *n* will lead to

$$\sum_{k=0}^{P} a_k r_h(l-k) = d_0 h^*(-l) \qquad -\infty < l < \infty$$

- Since h(-l)=0 for l>0,

$$\sum_{k=0}^{P} a_k r_h(l-k) = 0$$ and $$\sum_{k=0}^{P} a_k r_h(-k) = |d_0|^2$$

for l=1,…,P   for l=0

- Combining these two equations

$$\begin{bmatrix} r_h(0) & r_h^*(1) & \cdots & r_h^*(P) \\ r_h(1) & r_h(0) & \cdots & r_h^*(P-1) \\ \vdots & \vdots & \ddots & \vdots \\ r_h(P) & r_h(P-1) & \cdots & r_h(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_P \end{bmatrix} = \begin{bmatrix} |d_0|^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

- In matrix notations, it is the Yule-Walker equations.

$$\mathbf{R}_x \mathbf{a} = -\mathbf{r}_x$$

- Where $\mathbf{R}_x$ is the autocorrelation matrix, **a** is the all pole parameter vector and $\mathbf{r}_x$ is the vector of autocorrelations.
- So the first P+1 autocorrelation values determine the coefficients.
- Conversely, the coefficients determine the first P+1 autocorrelation values.

- Normalizing the autocorrelations by dividing by r(o),

$$\mathbf{Pa} = -\boldsymbol{\rho}$$

$$\boldsymbol{\rho} = [\rho(1)\ \rho(2)\ \cdots\ \rho(P)]^H$$

- It can be re-written as:

$$\mathbf{A}\boldsymbol{\rho} = -\mathbf{a}$$

$$\text{where } \langle \mathbf{A} \rangle_{ij} = a_{i-j} + a_{i+j},$$

- Given the coefficients, the autocorrelation can be obtained.

- Second Order all-pole model

$$H(z) = \frac{d_0}{1 + a_1 z^{-1} + a_2 z^{-2}} = \frac{d_0}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})}$$

  – Representing the a1 and a2 in terms of the poles

$$a_1 = -(p_1 + p_2)$$

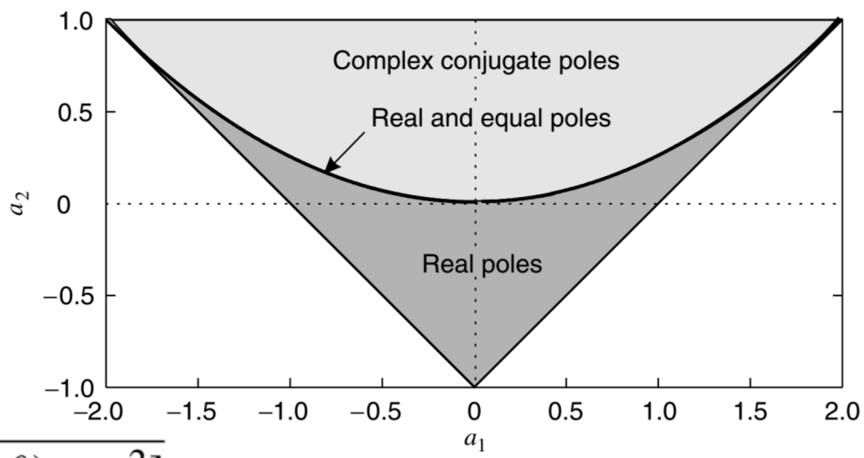
$$a_2 = p_1 p_2$$

  – To be stable and causal

$$-1 < a_2 < 1$$

$$a_2 - a_1 > -1$$

$$a_2 + a_1 > -1$$



$$h(n) = \frac{d_0}{p_1 - p_2} (p_1^{n+1} - p_2^{n+1}) u(n)$$

$$R(e^{j\omega}) = \frac{d_0^2}{[1 - 2r\cos(\omega - \theta) + r^2][1 - 2r\cos(\omega + \theta) + r^2]}$$

All-zero Models

- The output of the all-zero model is the weighted average of delayed versions of the input signal

$$x(n) = \sum_{k=0}^{Q} d_k w(n-k) \qquad H(z) = D(z) = \sum_{k=0}^{Q} d_k z^{-k}$$

- Its impulse response is:

$$h(n) = \begin{cases} d_n & 0 \le n \le Q \\ 0 & \text{elsewhere} \end{cases}$$

- Its autocorrelation is given as

$$r_h(l) = \sum_{n=-\infty}^{\infty} h(n)h^*(n-l) = \begin{cases} \sum_{k=0}^{Q-l} d_k d_{k+l}^* & 0 \le l \le Q \\ 0 & l > Q \end{cases}$$

- Note that these equations are nonlinear. So solution is complicated.
- The spectrum is

$$R_h(e^{j\omega}) = D(z)D(z^{-1})|_{z=e^{j\omega}} = |D(e^{j\omega})|^2 = \sum_{l=-Q}^{Q} r_h(l)e^{-j\omega l}$$

- Reading Assignment:
  - Low order model of all-zero models

Pole-Zero Models

- A general pole-zero model is given as

$$x(n) = -\sum_{k=1}^{P} a_k x(n-k) + \sum_{k=0}^{Q} d_k w(n-k)$$

- Its impulse response is given as

$$h(n) = -\sum_{k=1}^{P} a_k h(n-k) + d_n$$

- For n>Q, it can be shown that

$$h(n) = -\sum_{k=1}^{P} a_k h(n-k) \qquad n > Q$$

- Therefore, the first P+Q+1 values of the impulse response completely specify the pole-zero model.

- The autocorrelation can be represented as

$$\sum_{k=0}^{P} a_k r_h(l-k) = \sum_{k=0}^{Q} d_k h^*(k-l) \qquad \text{for all } l$$

- Noting that h(n) is causal,

$$\sum_{k=0}^{P} a_k r_h(l-k) = 0 \qquad l > Q$$

- Therefore, the a_k can be obtained from $l=Q+1,\dots,Q+P$

$$\begin{bmatrix} r_h(Q) & r_h(Q-1) & \cdots & r_h(Q-P+1) \\ r_h(Q+1) & r_h(Q) & \cdots & r_h(Q-P+2) \\ \vdots & \vdots & \ddots & \vdots \\ r_h(Q+P-1) & r_h(Q+P-2) & \cdots & r_h(Q) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix} = - \begin{bmatrix} r_h(Q+1) \\ r_h(Q+2) \\ \vdots \\ r_h(Q+P) \end{bmatrix}$$

- Note that this is not a Toeplitz matrix.
- Even after a_k are obtained, the solution to d_k is still nonlinear.
  - Reading Assignment:
    - Manolakis: pp 178-179
    - Hayes: pp 190 -193

Reading Assignment

- Models with poles on the unit circle.
- Cepstrum of pole-zero models.

Assignment 2

2.1 Implement the second order filter with a_1=0.6 and a_2=0.2 and drive it with IID(0,0.2). Then , obtain a sample realization of length 40. From this sample realization, obtain a second order deterministic all-pole model with Prony's method.

- Plot the obtained sample realization,
- Show the coefficients of the all-pole model obtained with Prony's method,
- Comment on the result.

2.2 Show that the autocorrelation matrix used to obtain the denominator coefficients in the pole-zero stochastic model is not Toeplitz.

2.3 Implement the autocorrelation and covariance methods by MATLAB. Record 10 seconds of your speech, break it into 32.5 millisecond windows and represent each of the windows by all pole model with P=14.

- The MATLAB code,
- The recorded speech,
- The model parameters and
- Discussion points.