Summary of Course-
Stochastic Analysis for Engineers

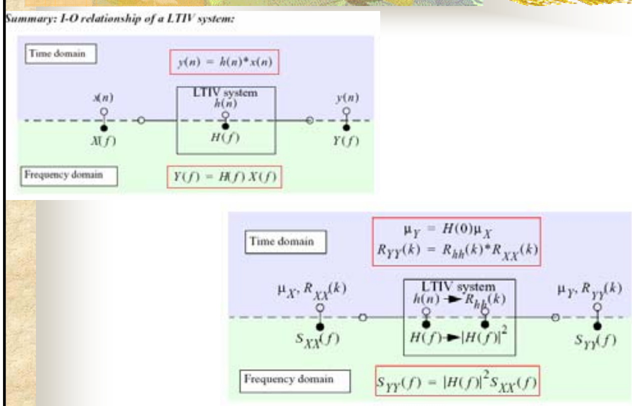
Zhenyu Yang
IRS7 Semester, Fall 2004, Aalborg University Esbjerg
http://www.cs.aue.auc.dk/~yang/course/stoc04.htm

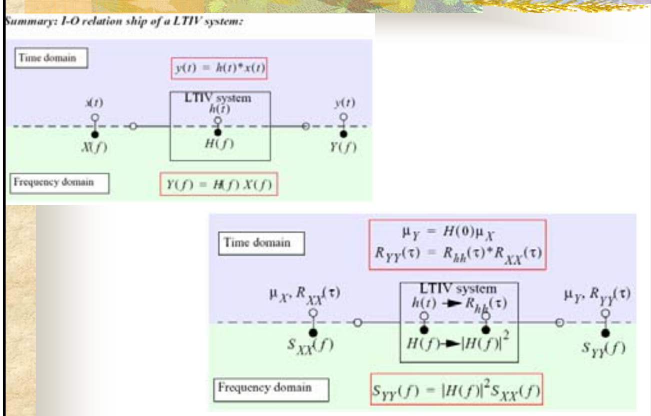
Objective

- To give students an **understanding** of the description of stochastic signals in order to perform filtering and detection
- To enable students to **apply** estimation and detection methods for simple problems in connection with stationary stochastic processes
- To give students an **understanding** of spectral estimation techniques

MM1. Response of LTI Systems to Random Inputs

Reading page: Chapt 4, pp.216-242





MM2. Discrete Linear Process Models

- Moving Average (MA) models
- Autoregressive (AR) models
- Autoregressive Moving Average (ARMA) models

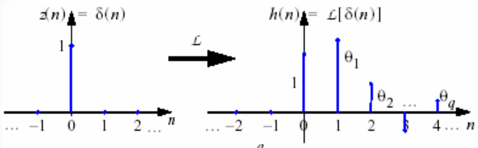
Reading page: Chapt 5, pp.250-275

Moving Average (MA) Processes

- Definition: A random sequence X[n] is a moving average process of order q (MA(q)) if for any n, there is

$$X[n] = Z[n] + \sum_{i=1}^{q} \theta_i Z[n-i] \qquad \theta_q \neq 0$$

Where Z[n] is a white Gaussian process



$$h(n) = \delta(n) + \sum_{\ell=1}^{q} \theta_\ell \delta(n-i)$$

Statistic Properties of MA Process

- Mean, autocorrelation and PSD functions of MA(q) process:

$$\mu_X = 0$$

$$R_{XX}[k] = \sigma_Z^2 R_{hh}[k] = \begin{cases} \sigma_Z^2[\theta_k + \sum_{j=k+1}^{q} \theta_j \theta_{j-k}] & k < q \\ \sigma_Z^2 \theta_q & k = q \\ 0 & k > q \end{cases}$$

$$where \quad R_{hh}[k] = h[k] * h[-k]$$

$$S_{XX}(f) = |1 + \sum_{i=1}^{q} \theta_i \exp(-j2\pi i f)|^2 \sigma_Z^2$$

See p.270

Autoregressive (AR) Models

- Definition: A random sequence X[n] is an autoregressive process of order p (AR(p)) if it is WSS and for any n, there is

$$X(n) = \sum_{i=1}^{p} \varphi_i X(n-i) + Z(n)$$

Where Z[n] is a white Gaussian process

- Recursive filters, all-pole models, state space model, ..

Statistic Properties of AR processes

- **Mean, autocorrelation and PSD functions of** AR process X[n]:

$$\mu_X = 0$$

$$R_{XX}[k] = \sigma_Z^2 R_{hh}[k]$$

$$S_{XX}(f) = \frac{\sigma_Z^2}{|1 - \sum_{i=1}^{p} \varphi_i \exp(-j2\pi i f)|^2}$$

Yule-Walker Equation

Inserting $k = 1, \ldots, p$ in (2.1) yields p identities that can be concatenated in a matrix form according to

$$\begin{bmatrix} R_{XX}(1) \\ R_{XX}(2) \\ \ldots \\ R_{XX}(p) \end{bmatrix} = \begin{bmatrix} R_{XX}(0) & R_{XX}(1) & \ldots & R_{XX}(p-1) \\ R_{XX}(-1) & R_{XX}(0) & \ldots & R_{XX}(p-2) \\ \ldots & \ldots & \ldots & \ldots \\ R_{XX}(-(p-1)) & R_{XX}(-(p-2)) & \ldots & R_{XX}(0) \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \ldots \\ \phi_p \end{bmatrix}$$

$$\gamma = \Gamma\Phi$$

System identification
Yule walker (autocorre.)

Comments:

- The feed-back coefficients $\phi_1, \ldots, \phi_p$ of the recursiv filter and the variance σ_Z^2 of the white Gaussian input process $Z(n)$ can be computed from $R_{XX}(0), \ldots, R_{XX}(p)$ via the Yule-Walker equations and vice-versa.
- The samples $R_{XX}(k)$, $k > p$ can be recursively computed from $\phi_1, \ldots, \phi_p$ and $R_{XX}(k-1), \ldots, R_{XX}(k-p)$ by using Identity (2.2).

Autoregressive Moving Average (ARMA) Models

- Definition: A random sequence X[n] is an autoregressive moving average (ARMA) process of order (p,q) (denoted as ARMA(p,q) if it is WSS and for any n, there is

$$X(n) = \sum_{i=1}^{p} \varphi_i X(n-i) + \sum_{j=1}^{q} \theta_j Z(n-j) + Z(n)$$

Where Z[n] is a white Gaussian process

- Any WSS process can be approximated by an ARMA(p,q) process

Properties of ARMA Processes

- Transfer function:

$$H(z) = \frac{1 + \sum_{k=1}^{q} \theta_k z^{-k}}{1 - \sum_{i=1}^{p} \varphi_i z^{-i}} \qquad H(f) = \frac{1 + \sum_{k=1}^{q} \theta_k e^{-j2k\pi f}}{1 - \sum_{i=1}^{p} \varphi_i e^{-j2i\pi f}}$$

- Stability and causality
- Statistic properties:

$$\mu_X = 0$$

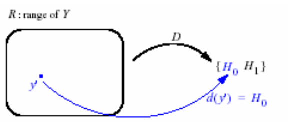
$$R_{XX}[k] = \sigma_Z{}^2 R_{hh}[k]$$

$$S_{XX}(f) = \frac{|1 + \sum_{k=1}^{q} \varphi_k \exp(-j2\pi k f)|^2}{|1 - \sum_{i=1}^{p} \varphi_i \exp(-j2\pi i f)|^2} \sigma_Z{}^2$$

MM3. Signal Detection (Part One)

- Hypothesis testing
- Decision rules
- Binary detection

Reading page: Chapt 6, pp.341-352

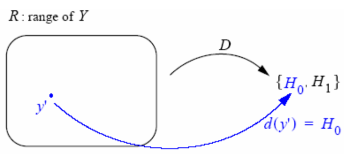


3.1 Hypothesis Testing

In hypothesis testing, a **decision** is made based on the **observation** of a random variable as to which of several **hypotheses** to accept

Binary Hypothesis Testing

- Two hypotheses: H_0 and H_1
- One observation y of a random variable Y whose pdf under each hypothesis is known, denoted as $f(y|H_0)$ and $f(y|H_1)$
- A decision rule is to decide
- $D: R \to \{H_0, H_1\}$, R range of Y



Decision Rule

Decision table:

Decision D	True hypothesis H: H_0	H_1
H_0	(H_0, H_0)	(H_0, H_1)
H_1	(H_1, H_0)	(H_1, H_1)

- (green) : Correct decision
- (red) : Incorrect decision:

Probability of correct decision:

$$P_c = P[D = H_0, H = H_0] + P[D = H_1, H = H_1] = P[D = H_0|H_0]P[H_0] + P[D = H_1|H_1]P[H_1]$$

Probability of incorrect decision:

$$P_e = P[D = H_1, H = H_0] + P[D = H_0, H = H_1] = P[D = H_1|H_0]P[H_0] + P[D = H_0|H_1]P[H_1]$$

Obviously,

$$P_c = 1 - P_e.$$

Types of error and their probability:

- **False alarm** (Type I error): $D = H_1$ when H_0 is true.

  False alarm probability:

$$P_f \equiv P[D = H_1|H_0]$$

- **Miss** (Type II error): $D = H_0$ when H_1 is true.

  Probability of a miss:

$$P_m \equiv P[D = H_0|H_1]$$

Example: BPAM Signal Detection

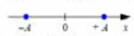
Example: Binary pulse amplitude modulation (BPAM)

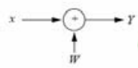
- Signal model:

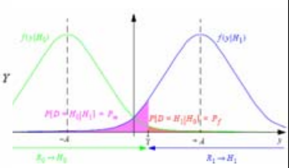
$$Y = x + W$$

- where

$$x = \begin{cases} -A & \text{under } H_0 \\ +A & \text{under } H_1 \end{cases}$$







- W is a Gaussian noise, i.e.:
  - W is a Gaussian random variable,
  - with expectation $\mathbf{E}[W] = 0$,
  - and variance $\mathbf{E}[W^2] = \sigma^2$.

Probability density function (pdf) of W:

$$f(w) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2} w^2\right\}$$

Pdf of Y under H_0 and H_1:

$$f(y|H_0) = f(w)|_{w=y+A} = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(y+A)^2\right\}$$

$$f(y|H_1) = f(w)|_{w=y-A} = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(y-A)^2\right\}$$

3.2 Decision Rules

- Maximum "a posteriori" decision rule
- Bayes decision rule – Costs of errors
- Minmax rule and Neyman-Pearson rule

Maximum "a posteriori" (MAP) Rule

- MAP decision rule:

$$f(y|H_1)P(H_1) \underset{H_0}{\overset{H_1}{\gtrless}} f(y|H_0)P(H_0)$$

- Prior distribution $P(H_i)$, i=1,2
- Likelihood ratio L(y):

$$L(y) = \frac{f(y|H_1)}{f(y|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P(H_0)}{P(H_1)}$$

$$l(y) = \ln(L(y))$$

$$l(y) = \ln\left(\frac{f(y|H_1)}{f(y|H_0)}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \ln\left(\frac{P(H_0)}{P(H_1)}\right)$$

Bayes' Decision Rule

- Average cost:

$$\overline{C} = C_{00}P[D=H_0|H_0]P[H_0] + C_{10}P[D=H_1|H_0]P[H_0] + C_{01}P[D=H_0|H_1]P[H_1] + C_{11}P[D=H_1|H_1]P[H_1]$$

- Bayes' decision rule: minimize the average cost

$$L(y) = \frac{f(y|H_1)}{f(y|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P(H_0)(C_{10}-C_{00})}{P(H_1)(C_{01}-C_{11})}$$

Example: BPAM MAP Detection (Cont'd)

- Likelihood ratio:

$$L(y) = \frac{f(y|H_1)}{f(y|H_0)} = \frac{\exp\left\{-\frac{1}{2\sigma^2}(y-A)^2\right\}}{\exp\left\{-\frac{1}{2\sigma^2}(y+A)^2\right\}} = \exp\left\{\frac{1}{2\sigma^2}[(y+A)^2-(y-A)^2]\right\}$$

- Loglikelihood ratio:

$$l(y) = \ln\left(\frac{f(y|H_1)}{f(y|H_0)}\right) = \frac{1}{2\sigma^2}[(y+A)^2-(y-A)^2] = \frac{2Ay}{\sigma^2}$$

- MAP decision rule:

$$\frac{2Ay}{\sigma^2} \underset{H_0}{\overset{H_1}{\gtrless}} \ln\left(\frac{P[H_0]}{P[H_1]}\right)$$

$$y \underset{H_0}{\overset{H_1}{\gtrless}} \frac{\sigma^2}{2A}\ln\left(\frac{P[H_0]}{P[H_1]}\right) = \tilde{\gamma}_{MAP}$$

Maximum likelihood (ML) decision rule:
Selecting a uniform "a priori" pdf for the hypotheses, i.e.

$$P[H_0] = P[H_1] = \frac{1}{2},$$

the MAP decision rule reduces to the ML decision rule:

$$f(y|H_1) \underset{H_0}{\overset{H_1}{\gtrless}} f(y|H_0)$$

MM4. Signal Detection (Part Two)

Reading page: Chapt 6, pp.352-361, 366-370

- 4.1 Binary detection of discrete-time signals
- 4.2 Binary detection of continuous-time signals
- 4.3 M-ary detection

4.1 Binary Discrete: Decision Rules

- Singal model: $Y(n)=x(n)+W(n)$, n=0,1,...N-1

$$x(n) = \begin{cases} s_0(n) & \text{under hypothesis } H_0 \\ s_1(n) & \text{under hypothesis } H_1 \end{cases}$$

$$\mathbf{Y} = [Y_0 \; Y_1 \; \cdots \; Y_{N-1}]^T \quad \mathbf{y} = [y_0 \; y_1 \; \cdots \; y_{N-1}]^T$$

$$\mathbf{S}_0 = [s_{00} \; s_{01} \; \cdots \; s_{0N-1}]^T \quad \mathbf{S}_1 = [s_{10} \; s_{11} \; \cdots \; s_{1N-1}]^T$$

$$\mathbf{W} = [W_0 \; W_1 \; \cdots \; W_{N-1}]^T$$

- W(n) is a white Guaussian noise:

$$W(n) \quad \text{is a Gaussian process}$$
$$E\{W(n)\} = 0$$
$$R_{WW}(k) = \sigma^2\delta(k)$$

- MAP decision rule:

$$\mathbf{y}^T(\mathbf{s_1}-\mathbf{s_0}) \underset{H_0}{\overset{H_1}{\gtrless}} \sigma^2 \ln\left(\frac{P(H_0)}{P(H_1)}\right) + \frac{1}{2}\left(E_{s_1} - E_{s_0}\right) \qquad E_{s_i} = \|\mathbf{s}_i\|^2 = \sum_{n=0}^{N-1} s_{in}^2$$

4.2 Binary Continuous: Decision Rules

- Time-limited but possibly bandwidth unlimited finite-energy signals
- Decision rule:

$$W(t) \quad \text{is a Gaussian process}$$
$$E\{W(t)\}=0$$
$$R_{WW}(\tau)=\frac{N_0}{2}\delta(\tau) \quad S_{WW}(f)=\frac{N_0}{2}$$

$$\int_{-T_s}^{T_e} y(t)(s_1(t)-s_0(t))dt \underset{H_0}{\overset{H_1}{\gtrless}} \frac{N_0}{2}\ln(\gamma)+\frac{1}{2}\left(E_{\mathbf{s}_1}-E_{\mathbf{s}_0}\right) \qquad E_{\mathbf{s}_i}=\|\mathbf{s}_i\|^2=\int_{T_s}^{T_e}(s_i(t))^2 dt$$

- MAP decision rule:

$$\int_{T_s}^{T_e} y(t)(s_1(t)-s_0(t))dt \underset{H_0}{\overset{H_1}{\gtrless}} \frac{N_0}{2}\ln\left(\frac{P(H_0)}{P(H_1)}\right)+\frac{1}{2}\left(E_{\mathbf{s}_1}-E_{\mathbf{s}_0}\right) \qquad E_{\mathbf{s}_i}=\|\mathbf{s}_i\|^2=\int_{T_s}^{T_e}(s_i(t))^2 dt$$

4.3 MAP for M-ary Decision

- MAP decision rule:

$$\text{select} \quad H_i \quad \text{if} \quad P(H_i\mid y) \ge P(H_j \mid y) \quad \text{for} \quad \text{any} \quad j=0,1,\cdots,M-1$$
$$\text{or}$$
$$\text{select} \quad H_i \quad \text{if} \quad \frac{f(y\mid H_i)}{f(y\mid H_j)} \ge \frac{P(H_j)}{P(H_i)} \quad \text{for} \quad \text{any} \quad j=0,1,\cdots,M-1$$

- MAP decision rule for time-limited discrete-time signals:

$$\text{select} \quad H_i \quad \text{if} \quad \mathbf{y}^{\mathbf{T}}\mathbf{s}_{\mathbf{i}}+\sigma^2\ln(P(H_i))-\frac{1}{2}E_{\mathbf{s}_i} \ge \mathbf{y}^{\mathbf{T}}\mathbf{s}_j+\sigma^2\ln(P(H_j))-\frac{1}{2}E_{\mathbf{s}_j} \quad \text{for} \quad \text{any} \quad j=0,1,\cdots,M-1$$

- Further with uniform"a priori" pdf, i.e., $P(H_0)=P(H_1)=\ldots=P(H_{M-1})=1/M$

$$\text{select} \quad H_i \quad \text{if} \quad \mathbf{y}^{\mathbf{T}}\mathbf{s}_{\mathbf{i}}-\frac{1}{2}E_{\mathbf{s}_i} \ge \mathbf{y}^{\mathbf{T}}\mathbf{s}_j-\frac{1}{2}E_{\mathbf{s}_j} \quad \text{for} \quad \text{any} \quad j=0,1,\cdots,M-1$$

MM5. Minimum Mean Squared Error Estimation

Reading page: Chapt 7, pp.377-397

- **5.1 Linear minimum mean squared error estimators**
- 5.2 (Nonlinear) minimum mean squared error estimator

5.1 Linear Minimum Mean Squared Error (LMMSE) Estimators

LMMSE Problem formulation

- A random sequence X(1), ... X(M) whose realization can be observed
- A random variable Y which has to be estimated
- Seek a linear estimator as:

$$\hat{Y}=h_0+\sum_{m=1}^{M}h_m X(m)$$

- By miminizing the mean squared error(MSE):

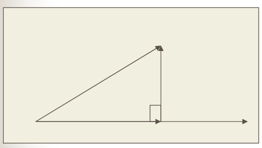
$$\min_{h_m, m=0,1,\ldots M} E\{(Y-\hat{Y})^2\}$$

5.1.1 Orthogonality Principle

- A necessary condition for a linear estimator denoted by $h=[h_0, \ldots h_M]^T$ to be the solution of the LMMSE is that

$$E\{Y-\hat{Y}\}=E\left\{Y-\left(h_0+\sum_{m=1}^{M}h_m X(m)\right)\right\}=0 \qquad E\{Y\}=E\{\hat{Y}\}$$

$$E\{(Y-\hat{Y})X(j)\}=E\left\{\left(Y-\left(h_0+\sum_{m=1}^{M}h_m X(m)\right)\right)X(j)\right\}=0, \quad j=1,\ldots,M$$



- Orthogonality Principle

5.1.3 LMMSE Soultion

$$\mathbf{h}=[h_1 \quad \cdots \quad h_M]^T \qquad \mu_{\mathbf{X}}=[\mu_{X(1)} \quad \cdots \quad \mu_{X(M)}]^T$$

$$\sum_{XY}=\begin{bmatrix}\sum_{X(1)Y}\\ \vdots \\ \sum_{X(M)Y}\end{bmatrix}, \qquad \sum_{\mathbf{XX}}=\begin{bmatrix}\sum_{X(1)X(1)} & \cdots & \sum_{X(1)X(M)}\\ \vdots & \ddots & \vdots \\ \sum_{X(M)X(1)} & \cdots & \sum_{X(M)X(M)}\end{bmatrix},$$

$$\mathbf{h}=\left(\sum_{\mathbf{XX}}\right)^{-1}\sum_{XY}$$
$$h_0=\mu_Y-\mathbf{h}^T\mu_{\mathbf{X}}=\mu_Y-\left(\sum_{XY}\right)^T\left(\sum_{\mathbf{XX}}\right)^{-1}\mu_{\mathbf{X}}$$

$$E\{(Y-\hat{Y})^2\}=E\{Y^2\}-E\{\hat{Y}^2\}=E\{(Y-\hat{Y})Y\}$$
$$E\{(Y-\hat{Y})^2\}=\sigma_Y{}^2-\mathbf{h}^T\sum_{XY}$$

5.1.6 Example

- *Example: Linear prediction of a WSS process*

Let $Y(n)$ denote a WSS process with

- zero mean, i.e $\mathbf{E}[Y(n)] = 0$,
- autocorrelation function $\mathbf{E}[Y(n)Y(n+k)] = R_{YY}(k)$

We seek the LMMSEE for the present value of $Y(n)$ based on the M past observations $Y(n-1), \dots, Y(n-M)$ of the process. Hence,

- $Y = Y(n)$
- $X(m) = Y(n-m)$, $m = 1, \dots, M$, i.e.

$$X = [Y(n-1), \dots, Y(n-M)]^T$$

Because $\mu_Y = 0$ and $\mu_X = 0$, it follows from (3.4*b*) that

$$h_0 = 0$$

Computation of Σ_{XY} and Σ_{XX}:

- $\Sigma_{XY} = [\mathbf{E}[Y(n-1)Y(n)], \dots, \mathbf{E}[Y(n-M)Y(n)]]^T = [R_{YY}(1), \dots, R_{YY}(M)]^T$
- $\Sigma_{XX} =$

$$= \begin{bmatrix} \mathbf{E}[Y(n-1)^2] & \mathbf{E}[Y(n-1)Y(n-2)] & \dots & \mathbf{E}[Y(n-1)Y(n-M)] \\ \mathbf{E}[Y(n-2)Y(n-1)] & \mathbf{E}[Y(n-2)^2] & \dots & \mathbf{E}[Y(n-2)Y(n-M)] \\ \dots & \dots & \dots & \dots \\ \mathbf{E}[Y(n-M)Y(n-1)] & \mathbf{E}[Y(n-M)Y(n-2)] & \dots & \mathbf{E}[Y(n-M)^2] \end{bmatrix}$$

$$= \begin{bmatrix} R_{YY}(0) & R_{YY}(1) & R_{YY}(2) & \dots & R_{YY}(M-1) \\ R_{YY}(1) & R_{YY}(0) & R_{YY}(1) & \dots & R_{YY}(M-2) \\ R_{YY}(2) & R_{YY}(1) & R_{YY}(0) & \dots & R_{YY}(M-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_{YY}(M-1) & R_{YY}(M-2) & R_{YY}(M-3) & \dots & R_{YY}(0) \end{bmatrix}$$

5.2 Minimum Mean Squared Error Estimators (MMSEE)

- **MMSEE:** The solution for MMSEE of Y based on the observation of X(1),....,X(M) is:

$$\tilde{Y}(X(1),\dots,X(M)) = E\{Y \mid X(1),\dots,X(M)\}$$

- Which reaches $\min E\{(Y-\tilde{Y})^2\}$
- Specially, if X(1)=x(1),....,X(M)=x(M) is observaed, then

$$\tilde{Y}(x(1),\dots,x(M)) = E\{Y \mid x(1),\dots,x(M)\} = \int y p(y \mid x(1),\dots,x(M))dy$$

- $\tilde{Y} = \hat{Y}$ iff [Y X(1),....,X(M)] is a Gaussian vector

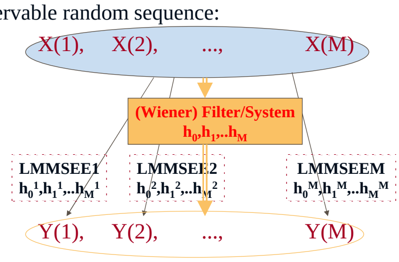
MM6. Discrete-Time Wiener Filters

Reading page: Chapt 7, pp.406-419

- 6.1 Noncausal Wiener Filters
- 6.2 Causal Wiener Filters

6.0 An Intuitive Explanation

- Observable random sequence:



- A random sequence which needs to be estimated

Discrete-Time Wiener Filters

Motivation:

- Estimate a WSS random sequence Y(n) based on the observation of another sequence X(n).
- **Without loss of generality we assume that**
  E{Y(n)}=E{X(n)}=0
- The goodness of the estimator is described by MSE

$$E\{(Y(n)-\hat{Y}(n))^2\}$$

6.1 Ideal (Noncausal) Wiener Filters

- **Problem Formulation:**

Seek a linear filter ← system

$$\hat{Y}(n) = \sum_{m=-\infty}^{\infty} h(m)X(n-m) = h(n) * X(n)$$

LMMSEE: $\hat{Y} = h_0 + \sum_{m=1}^{M} h_m X(m)$

↑ Signals' relationship

Which minimizes the MSE

$$E\{(\hat{Y}(n)-Y(n))^2\}$$

The filter reaching above requirement is called *ideal (noncausal) Wiener filter*

6.1.2 Wiener Filter in Transfer Function

Wiener-Hopf equation:

$$R_{XY}(k) = \sum_{m=-\infty}^{\infty} h(m) R_{XX}(k-m) = h(k) * R_{XX}(k)$$

- TF of the Wiener filter:

$$H(f) = \frac{S_{XY}(f)}{S_{XX}(f)}$$

- MSE residual:

$$E\{(Y(n)-\hat{Y}(n))^2\} = \sigma_Y^{\ 2} - \sum_{m=-\infty}^{\infty} h(m) R_{XY}(m)$$

$$E\{(Y(n)-\hat{Y}(n))^2\} = \int_{-\frac{1}{2}}^{\frac{1}{2}} [S_{YY}(f) - \frac{|S_{XY}(f)|2}{S_{XX}(f)}] df$$

See p.408-409 for proof

6.2.1 Causal Wiener Filter (Case A)

Case A: X(n) is a white noise with unity variance E{X(n)X(n+k)}=δ(k)

- Ideal Wiener filter:

$$\hat{Y}(n) = \sum_{m=-\infty}^{\infty} h(m) X(n-m) = h(n) * X(n)$$

- **A causal Wiener filter** can be achieved by cancelling the noncausal part of ideal Wiener filter:

$$\hat{Y}_c(n) = \sum_{m=0}^{\infty} h(m) X(n-m)$$

- The causal Wiener filter minimizes the MSE within the class of causal linear estimators

6.2.7 Finite Wiener Filter

$$Finite\ Wiener\ filter: \quad \hat{Y}(n) = \sum_{m=0}^{M} h(m) X(n-m)$$

Wiener – Hopf Solution:

$$\mathbf{h}^{\mathbf{T}} = [h(0)\ h(1) \cdots h(M)] = \mathbf{R}_{\mathbf{XX}}^{\ -1} \mathbf{R}_{\mathbf{XY}}$$

provided $\mathbf{R}_{\mathbf{XX}}$ *is invertible,*

where

$$\mathbf{R}_{\mathbf{XX}} = \begin{bmatrix} R_{XX}(0) & R_{XX}(1) & \cdots & R_{XX}(M) \\ R_{XX}(1) & R_{XX}(0) & \cdots & R_{XX}(M-1) \\ \vdots & \vdots & \ddots & \vdots \\ R_{XX}(M) & R_{XX}(M-1) & \cdots & R_{XX}(0) \end{bmatrix}$$

$$\mathbf{R}_{\mathbf{XY}}^{\ \mathbf{T}} = [R_{XY}(0)\ R_{XY}(1) \cdots R_{XY}(M)]$$

- LMMSE:

$$\mathbf{h} = [h_1 \ \cdots \ h_M]^T \qquad \mu_{\mathbf{X}} = [\mu_{\mathbf{X}(1)} \ \cdots \ \mu_{\mathbf{X}(M)}]^T$$

$$\sum_{\mathbf{X}Y} = \begin{bmatrix} \sum_{X(1)Y} \\ \vdots \\ \sum_{X(M)Y} \end{bmatrix}, \qquad \sum_{\mathbf{XX}} = \begin{bmatrix} \sum_{X(1)X(1)} & \cdots & \sum_{X(1)X(M)} \\ \vdots & \ddots & \vdots \\ \sum_{X(M)X(1)} & \cdots & \sum_{X(M)X(M)} \end{bmatrix},$$

$$\mathbf{h} = \left(\sum_{\mathbf{XX}}\right)^{-1} \sum_{\mathbf{X}Y}$$

$$h_0 = \mu_Y - \mathbf{h}^T \mu_{\mathbf{X}} = \mu_Y - \left(\sum_{\mathbf{X}Y}\right)^T \left(\sum_{\mathbf{XX}}\right)^{-1} \mu_{\mathbf{X}}$$

- MSE residual:

$$E\{(Y-\hat{Y})^2\} = E\{Y^2\} - E\{\hat{Y}^2\} = E\{(Y-\hat{Y})Y\}$$

$$E\{(Y-\hat{Y})^2\} = \sigma_Y^{\ 2} - \mathbf{h}^T \sum_{\mathbf{X}Y}$$

MM7, 8. Kalman Filter

Reading page: Chapt 7, pp.406-419

- **7.1 Introduction**
- 7.2 An Intuitive Description of Kalman filter
- 7.3 Formal Description of Scale Kalman Filter

7.1.1 What's Kalman Filter?

- One of the most well-known and often-used math. Tools for stochastic estimation from noisy measurements
- Rudolph E. Kalman in 1960 published his famous paper decribing a recursive solution to discrete-time linear filtering problem

Features

- Just some applied mathematics
- A linear system
- Noisy data in → hopefully less noisy output
- Delay is the price for filtering

7.2 Scalar Kalman Filter

- **System model**

$\{Y(n)\}$ is an unobservable sequence which is described by the following ***state or system equation***:

$$Y(n) = h(n)Y(n-1) + Z(n), n = 1, 2, \ldots \qquad (5.1)$$

- **Observation (or channel) model**

The observable sequence $X(n)$ is given by

$$X(n) = a(n)Y(n) + W(n) \quad , n = 1, 2, \ldots \qquad (5.2)$$

Property of the noise $\{W(n)\}$:

$\{W(n)\}$ is a non-stationary white noise.

- $E[W(n)] = 0$
- $E[W(n)W(n+k)] = \sigma_W^2(n)\delta(k)$

Initialization:

$Y(0)$ is a random variable whose expectation $\mu_{Y(0)} \equiv E[Y(0)]$ and variance $\sigma_{Y(0)}^2 \equiv E[(Y(0) - \mu_{Y(0)})^2]$ are known.

7.2 Scalar Kalman Filter (Cont'd)

Recursive implementation of the LMMSEE of $Y(n)$:

$$\underbrace{\hat{Y}(n+1 \mid n+1)}_{\substack{\text{Estimation} \\ \text{at time } n+1}} = \mathcal{LF}\Big(\underbrace{\hat{Y}(n \mid n)}_{\substack{\text{Estimation} \\ \text{at time } n}}, \underbrace{X(n+1)}_{\substack{\text{Observation} \\ \text{at time } n+1}}\Big)$$

The recursion is split into two steps:

- Step 1: ***One-step prediction***:

$$P : \hat{Y}(n \mid n) \underset{P}{\rightarrow} \hat{Y}(n+1 \mid n)$$

- Step 2: ***Updating***:

$$U : \hat{Y}(n+1 \mid n) \xrightarrow[U]{X(n+1)} \hat{Y}(n+1 \mid n+1)$$

7.2 Scalar Kalman Filter (Cont'd)

- **Prediction step:**

$$\hat{Y}(n+1 \mid n) = h(n+1)\hat{Y}(n \mid n) \quad (5.3)$$

$$R(n+1 \mid n) = h^2(n+1)R(n \mid n) + \sigma^2_{ZZ}(n+1) \quad (5.4)$$

- **Updating step:**

$$\hat{Y}(n+1 \mid n+1) = \hat{Y}(n+1 \mid n) + b(n+1)[X(n+1) - \hat{X}(n+1 \mid n)] \quad (5.5)$$

$$R(n+1 \mid n+1) = [1 - b(n+1)a(n+1)]R(n+1 \mid n) \quad (5.6)$$

with

$$b(n+1) \equiv \frac{a(n+1)R(n+1 \mid n)}{a(n+1)^2 R(n+1 \mid n) + \sigma^2_{WW}(n+1)}$$

- **Initialization:**

$$\hat{Y}(0 \mid 0) = \mu_{Y(0)}$$

$$R(0 \mid 0) = \sigma^2_{Y(0)}$$

MM8. Vector Kalman Filter

- **System model**

$$\mathbf{Y}(n) = \mathbf{H}(n)\mathbf{Y}(n-1) + \mathbf{Z}(n), \quad n = 1, 2, \ldots \quad (5.10)$$

where:

- $\mathbf{Y}(n) = [Y_1(n), \ldots, Y_r(n)]^T$: r-dimensional (r-D) random vector.
- $\{\mathbf{Z}(n)\}$: r-D non-stationary white noise vector:
  - $E[\mathbf{Z}(n)] = \mathbf{0}$
  - $\Sigma_{\mathbf{Z}(n)\mathbf{Z}(n+k)} = \mathbf{Q}_{\mathbf{Z}}(n)\delta(k)$
- $\{\mathbf{H}(n)\}$: sequence of known $r \times r$ matrices.

- **Observation Model**

$$\mathbf{X}(n) = \mathbf{A}(n)\mathbf{Y}(n) + \mathbf{W}(n), n = 1, 2, \ldots \quad (5.11)$$

where:

- $\mathbf{X}(n) = [X_1(n), \ldots, X_s(n)]^T$: s-D random vector.
- $\{\mathbf{W}(n)\}$: s-D non-stationary white noise vector with auto-covariance $\Sigma_{\mathbf{W}(n)\mathbf{W}(n+k)} = \mathbf{Q}_{\mathbf{W}(n)}\delta(k)$
- $\{\mathbf{A}(n)\}$: sequence of known $s \times r$ matrices.

Initialization

$\mathbf{Y}(0)$ is a random vector specified by its expectation $\mu_{\mathbf{Y}(0)}$ and covariance matrix $\Sigma_{\mathbf{Y}(0)\mathbf{Y}(0)}$.

- **Additional independence assumption**

$\mathbf{Y}(0)$, $\{\mathbf{Z}(n)\}$, and $\{\mathbf{W}(n)\}$ are uncorrelated.

MM8. Vector Kalman Filter (Cont'd)

Let us define

- $\hat{\mathbf{Y}}(n \mid n) \equiv$ LMMSEE of $\mathbf{Y}(n)$ based on $\mathbf{X}(1), \ldots, \mathbf{X}(n)$
- $\hat{\mathbf{Y}}(n+1 \mid n) \equiv$ LMMSEE of $\mathbf{Y}(n+1)$ based on $\mathbf{X}(1), \ldots, \mathbf{X}(n)$
- $\hat{\mathbf{X}}(n+1 \mid n) \equiv$ LMMSEE of $\mathbf{X}(n+1)$ based on $\mathbf{X}(1), \ldots, \mathbf{X}(n)$
- $\mathbf{R}(n \mid n) \equiv E[(\mathbf{Y}(n) - \hat{\mathbf{Y}}(n \mid n))(\mathbf{Y}(n) - \hat{\mathbf{Y}}(n|n))^T]$
- $\mathbf{R}(n+1 \mid n) \equiv E[(\mathbf{Y}(n+1) - \hat{\mathbf{Y}}(n+1 \mid n))(\mathbf{Y}(n+1) - \hat{\mathbf{Y}}(n+1|n))^T]$

Initialization :

$$\hat{\mathbf{Y}}(0 \mid 0) = \mu_{\mathbf{Y}}(0)$$

$$\mathbf{R}(0 \mid 0) = \Sigma_{\mathbf{Y}(0)\mathbf{Y}(0)}$$

MM8. Vector Kalman Filter (Cont'd)

- **Recursive equations of the Kalman filter**

Prediction Step :

$$\hat{\mathbf{Y}}(n+1 \mid n) = \mathbf{H}(n+1)\hat{\mathbf{Y}}(n \mid n)$$

$$\hat{\mathbf{X}}(n+1 \mid n) = \mathbf{A}(n+1)\hat{\mathbf{Y}}(n+1 \mid n)$$

$$\mathbf{R}(n+1 \mid n) = \mathbf{H}(n+1)\mathbf{R}(n \mid n)\mathbf{H}(n+1)^T + \mathbf{Q}_{\mathbf{Z}}(n+1)$$

Updating Step :

$$\hat{\mathbf{Y}}(n+1 \mid n+1) = \hat{\mathbf{Y}}(n+1 \mid n) + \mathbf{B}(n+1)[\mathbf{X}(n+1) - \hat{\mathbf{X}}(n+1 \mid n)]$$

$$\mathbf{R}(n+1 \mid n+1) = [\mathbf{I} - \mathbf{B}(n+1)\mathbf{A}(n+1)]\mathbf{R}(n+1 \mid n)$$

with the Kalman matrix

$$\mathbf{B}(n+1) = \mathbf{R}(n+1 \mid n)\mathbf{A}(n+1)^T[\mathbf{A}(n+1)\mathbf{R}(n+1 \mid n)\mathbf{A}(n+1)^T + \mathbf{Q}_{\mathbf{W}}(n+1)]^{-1}$$

MM9, MM10 Model-Free and Spectral Estimation of Random Processes

Model-Free Mean Estimation

In this section $\{X(n)\}$ is a WSS process with

- mean value: $\mu_X \equiv \mathrm{E}[X(n)]$
- autocorrelation function: $R_{XX}(k) \equiv \mathrm{E}[X(n)X(n+k)]$

The autocovariance function of $\{X(n)\}$ is

$$C_{XX}(k) \equiv \mathrm{E}[(X(n)-\mu_X)(X(n+k)-\mu_X)] = R_{XX}(k)-\mu_X^2$$

Observed sequence:

We assume that $\{X(0), \ldots, X(N-1)\}$ can be observed.

6.1.1. Estimation of the mean-value

- ***Arithmetic mean:***

$$\hat{\mu}_X = \overline{X} = \frac{1}{N}\sum_{n=0}^{N-1} X(n)$$

- ***Mean and variance of $\overline{X}$:***
  - Mean: $\overline{X}$ is an unbiased estimator of μ_X:

$$\mu_{\overline{X}} = \mu_X$$

  - Variance:

$$\sigma_{\overline{X}}^2 = \frac{1}{N}\sum_{k=-(N-1)}^{N-1}\left[1-\frac{|k|}{N}\right]C_{XX}(k)$$

Special case: When $\{X(n)\}$ is an uncorrelated process:

$$\sigma_{\overline{X}}^2 = \frac{1}{N}C_{XX}(0) = \frac{1}{N}\sigma_X^2$$

Model-Free Autocorrelation Estimation

6.1.2. Estimation of the autocorrelation function:

- ***Biased sample autocorrelation function:***

$$\hat{R}_{XX}(k) \equiv \begin{cases} \frac{1}{N}\sum_{n=0}^{N-k-1} X(n)X(n+k) \ ; & k = 0, \ldots, N-1 \\ \hat{R}_{XX}(-k) & ; \ k = -(N-1), \ldots, -1 \\ 0 & ; \ |k| \geq N \end{cases}$$

With this definition, the bias of $\hat{R}_{XX}(k)$ can be recast as

$$\mathrm{E}[\hat{R}_{XX}(k)] = w_B(k)R_{XX}(k)$$

The function

$$w_B(k) \equiv \frac{1}{N}R_{gg}(k) = \begin{cases} 1-\frac{|k|}{N} \ ; & |k| < N \\ 0 \ ; & \text{otherwise} \end{cases}$$

is called the **Bartlett window**.

- ***Unbiased sample autocorrelation function:***

$$\widehat{R}_{XX}(k) \equiv \begin{cases} \frac{1}{N-k}\sum_{n=0}^{N-k-1} X(n)X(n+k) \ ; & k = 0, \ldots, N-1 \\ \widehat{R}_{XX}(-k) & ; \ k = -(N-1), \ldots, -1 \\ 0 & ; \ |k| \geq N \end{cases}$$

$\widehat{R}_{XX}(k)$ is unbiased for $|k| < N$:

$$\mathrm{E}[\widehat{R}_{XX}(k)] = w_r(k)R_{XX}(k)$$

Model-Free PDF Estimation

The periodogram of $X_{obs}(n)$ is defined to be

$$\hat{S}_{XX}(f) = \mathcal{F}\{\hat{R}_{XX}(k)\} = \frac{1}{N}\left|\sum_{n=0}^{N-1} X(n)\exp(-j2\pi nf)\right|^2 = \frac{1}{N}|\mathcal{F}\{X_{obs}(n)\}(f)|^2 \qquad f \in [0, 1)$$

the discrete-frequency periodogram can be computed as

$$\hat{S}_{XX}(m) \equiv |\mathcal{F}_d\{X_{obs}(n)\}(m)|^2$$

the bias of $\hat{S}_{XX}(f)$ and $\hat{S}_{XX}(m)$ are given by

$$\mathrm{E}[\hat{S}_{XX}(f)] = W_F(f) * S_{XX}(f)$$
$$\mathrm{E}[\hat{S}_{XX}(m)] = [W_F(f) * S_{XX}(f)]\big|_{f=m/N}$$

The Fourier transform

$$W_F(f) \equiv \mathcal{F}\{w_B(k)\} = \frac{1}{N}\left(\frac{\sin(\pi fN)}{\sin(\pi f)}\right)^2$$

of the Bartlett window is called the Féjer kernel.

Model-Based Estimation

6.2. Parametric (model-based) estimation of the autocorrelation and spectrum

6.2.1. Box-Jenkins method:

- ***Key idea of the method:***
  - The observed sequence $\{y(0), \ldots, y(N'-1)\}$ is transformed in such a way that the transformed sequence $\{x(0), \ldots, x(N-1)\}$ can be reasonably assumed to be the realization of a WSS process $\{X(n)\}$.
  - An ARMA*(p,q)* process is fitted to $\{x(0), \ldots, x(N-1)\}$.
  - The estimated autocorrelation function and power spectrum are identified to the autocorrelation function and the power spectrum of the estimated ARMA*(p,q)* process.

steps of the Box-Jenkins method: