2. Optimum Linear Filters

2.1 Optimum Signal Estimation

Let:
- \( y(n) \) be the value of the signal of interest (or desired response) at time \( n \)
- \( x_k(n) \) be the set of \( M \) values (observations or data), for \( 1 \leq k \leq M \), at time \( n \)
  - signal of the \( k \)th sensor at time \( n \) (array processing)
  - \( x(n-k) \), the \( k \)th delay of the signal at time \( n \) (signal prediction, system inversion and identification)

then:

Given the set of data, \( x_k(n) \), the signal estimation problem is to determine an estimate \( \hat{y}(n) \) of the desired response, \( y(n) \) using:
\[
\hat{y}(n) \equiv H\{x_k(n), \ 1 \leq k \leq M\}
\]
where in the case of \( x_k(n) = x(n-k) \), the estimator takes the form of a discrete-time filter. We want to find an optimum estimator that approximates the desired response as closely as possible according to certain performance criterion, most commonly minimisation of the error signal:
\[
e(n) = y(n) - \hat{y}(n)
\]

2.2 Linear Mean Square Error Estimation

Design an estimator that provides an estimate \( \hat{y}(n) \) of the desired response \( y(n) \) using a linear combination of the data \( x_k(n) \) for \( 1 \leq k \leq M \), such that the MSE \( E[|y(n) - \hat{y}(n)|^2] \) is minimised. That is, our linear MSE estimator (dropping the time index \( n \)) is defined by:
\[
\hat{y} = \sum_{k=1}^{M} c_k x_k = c^T x = x^T c
\]

Equation 2.1

where:
\[
x = [x_1, x_2, \ldots, x_M]^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix}, \text{ is the } M \times 1 \text{ data or observation vector,}
\]
\[
c = [c_1, c_2, \ldots, c_M]^T = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{bmatrix}, \text{ is the } M \times 1 \text{ parameter or co-efficient vector of the estimator}
\]

and we want to derive the Linear Minimum Mean Square Error (LMMSE) estimator given by:
\[
c_o = \arg \min_c P(c), \text{ where } P(c) = E[|e|^2] = E[|y - \hat{y}|^2]
\]

and form the LMMSE estimate, \( \hat{y}_o = c_o^T x \)
2.2.1 Derivation of Linear MMSE Estimator

We first derive an expression for \( P(c) \) using Equation 2.1:

\[
P(c) = E[ |e|^2 ] = E[ (y - \hat{y})(y - \hat{y}) ] \\
= E[ (y - c^T x)(y - c^T x) ] \\
= E[ |y|^2 ] - c^T E(xy) - E(xy^T) c + c^T E(xx^T) c \\
= P_y - c^T d - d^T c + c^T R c
\]

where:

- \( P(c) \), is the MSE which we want to minimise
- \( P_y = E[ |y|^2 ] \), is the power of the desired response
- \( d = E(xy) \), is the \( M \times 1 \) cross-correlation vector between the data and desired response
- \( R = E\{xx^T\} \), is the \( M \times M \) correlation matrix of the data vector \( x \)

- The correlation matrix \( R \) is guaranteed to be symmetric (\( R^T = R \)) and nonnegative definite and in practice is positive definite. (i.e. \( x^T Rx > 0 \), for every non-zero \( x \))
- If \( R \) is positive definite, then \( R^{-1} \) exists and is also positive definite (i.e. \( R^{-T} = R^{-1} \) and \( x^T R^{-1} x > 0 \))

We next rewrite the expression for \( P(c) \) in the form of a “perfect square” as follows:

\[
P(c) = P_y - d^T R^{-1} d + (Rc - d)^T R^{-1} (Rc - d)
\]

**Exercise 2.1** Show This!

\( P(c) \) is minimised when \( Rc - d = 0 \) (Why?)

The necessary and sufficient conditions that determine the linear MMSE estimator, \( c_O \), are:

\[
Rc_O = d
\]

which can be written as the set of normal equations:

\[
\begin{bmatrix}
  r_{11} & r_{12} & \cdots & r_{1M} \\
  r_{21} & r_{22} & \cdots & r_{2M} \\
  \vdots & \vdots & \ddots & \vdots \\
  r_{M1} & r_{M2} & \cdots & r_{MM}
\end{bmatrix}
\begin{bmatrix}
  c_1 \\
  c_2 \\
  \vdots \\
  c_M
\end{bmatrix}
=
\begin{bmatrix}
  d_1 \\
  d_2 \\
  \vdots \\
  d_M
\end{bmatrix}
\]

Equation 2.2

where \( r_{ij} = E\{x_ix_j\} \) and \( d_i = E\{x_iy\} \).

A solution is guaranteed to exist if \( R \) is positive definite (i.e. \( c_O = R^{-1}d \)) and any general-purpose routine for the solution of simultaneous linear equations can be used.

The MMSE \( P_O = E[ |e_O|^2 ] \) where \( e_O = y - \hat{y}_O \) is:

\[
P_O = P_y - d^T R^{-1} d = P_y - d^T c_O
\]
We note:

1. We assume that $x$ and $y$ are zero mean (i.e. uncorrelated implies orthogonal). If not, we replace $x$ and $y$ by $x - E(x)$ and $y - E(y)$ respectively.
2. If $x$ and $y$ are uncorrelated then $d = 0$ and $P_O = P_x$, which means that there is no linear estimator that can be found to reduce the MSE since the desired response $y$ is uncorrelated with the data vector $x$. This is the worst result.
3. If $y$ can be perfectly estimated from $x$ then $\hat{y} = y$ and $P_O = 0$. The is the best result.

### 2.2.2 Principle of Orthogonality

We consider the correlation between the data vector $x$ and the MSE error, $e_O$:

$$E(x e_o) = E(x(y - x^T c_o)) = E(xy) - E(x x^T) c = d - Re_o = 0$$

that is:

$$E(x e_o) = 0$$

or:

$$E(x_i e_o) = 0 \quad \text{for } 1 \leq i \leq M$$

**Orthogonality Principle**

The estimation error, $e_O$, is orthogonal to the data, $x$, used for the estimation, i.e. $E(x e_o) = 0$

A convenient way to view this is to consider the abstract vector space where a vector is a zero-mean random variable with the following associations:

$$\|x\|^2 := x, x \geq E(|x|^2), \text{ for the length of the vector}$$

$$\|x\|_y \cos \theta_{xy} := x, y \geq E(xy), \text{ for the relative direction of a vector}$$

We illustrate this principle for the case of $M=2$ in [Figure 2-1](#).

![Figure 2-1](#) Illustration of orthogonality principle (Figure 6.9 [1]). Note that $x_i, e_o$, $1 \leq i \leq M$, but it is not true that $x_i \perp x_j$ unless the data itself is uncorrelated

**Exercise 2.2** Show that $E(y e_o) = E(|e_o|^2) = P_o$, by noting that $\hat{y}_o$ is in the subspace spanned by $x$ and is thus orthogonal to $e_o$
2.3 Solution of the Normal Equations

A numerical method for solution of the normal equations (Equation 2.2) and computation of the minimum error is known as the lower-diagonal-upper decomposition or LDL^T decomposition for short where the correlation matrix is written as: \( R = LDL^T \).

Full details of the LDL solution method can be found in [1, pgs 274-278].

2.4 Optimum Finite Impulse Response Filters (Wiener filters)

The optimum FIR filter (also known as Wiener filter) forms an estimate \( \hat{y}(n) \) of the desired response, \( y(n) \), by using finite samples from a related input signal, \( x(n) \). That is:

\[
\hat{y}(n) = \sum_{k=0}^{M-1} h(n,k)x(n-k) = \sum_{k=0}^{M-1} c_k(n)x(n-k) = c^T(n)x(n)
\]

\[\text{Equation 2.5}\]

where:
- \( c(n) = [c_0(n) \ c_1(n) \ldots \ c_{M-1}(n)]^T = [h(n,0) \ h(n,1) \ldots \ h(n,M-1)]^T \) is the filter impulse response sequence or coefficient vector at time \( n \)
- \( x(n) = [x(n) \ x(n-1) \ldots \ x(n-(M-1))]^T \) is the input data vector or filter memory at time \( n \)

We form the LMMSE estimate by solving the corresponding set of normal equations for the filter coefficients, \( c_o(n) \), at each time \( n \):

\[
R(n)c_o(n) = d(n)
\]

or in matrix form:

\[
\begin{bmatrix}
    r_{0,0}(n) & r_{0,1}(n) & \cdots & r_{0,M-1}(n) \\
    r_{1,0}(n) & r_{1,1}(n) & \cdots & r_{1,M-1}(n) \\
    \vdots & \vdots & \ddots & \vdots \\
    r_{M-1,0}(n) & r_{M-1,1}(n) & \cdots & r_{M-1,M-1}(n)
\end{bmatrix}
\begin{bmatrix}
    c_0(n) \\
    c_1(n) \\
    \vdots \\
    c_{M-1}(n)
\end{bmatrix}
= 
\begin{bmatrix}
    d_0(n) \\
    d_1(n) \\
    \vdots \\
    d_{M-1}(n)
\end{bmatrix}
\]

where \( R(n) = E\{x(n)x^T(n)\} \), \( r_{ij}(n) = E\{x(n-i)x(n-j)\} \), \( d(n) = E\{x(n)y(n)\} \), \( d_i(n) = E\{x(n-i)y(n)\} \) and then computing the estimate \( \hat{y}(n) \) using a discrete-time filter structure based upon Equation 2.5 primed by the \( c_o(n) \), as shown in Figure 2-2.
• The FIR filter coefficients have to be estimated and loaded into the filter at each sample time \( n \)
• Although the desired response \( y(n) \) is not available, the known relationship between \( y(n) \) and \( x(n) \) can be used to derive or estimate the cross-correlation vector \( d(n) = E\{x(n)y(n)\} \)

**Example 2–1**
The most common use of optimum filtering is for estimating a signal in noise, that is:

\[ x(n) = y(n) + v(n) \]

where \( v(n) \) is a noise signal which is assumed uncorrelated with \( y(n) \). Thus:

\[ d_i(n) = E\{x(n-k)y(n)\} = E\{(y(n-k) + v(n-k))y(n)\} = E\{y(n-k)y(n)\} \]

requiring only knowledge of the second-order statistics of the desired response \( y(n) \)

2.4.1 Optimum FIR Filters for Stationary Processes

Most useful FIR filters are designed when the input and desired response stochastic processes are jointly wide-sense stationary (WSS), in which case the correlation matrix, \( R(n) = R \), and cross-correlation vector, \( d(n) = d \), no longer depend explicitly on the time-index \( n \). That is for WSS processes:

\[ r_{xy}(n) = E\{x(n-i)x(n-j)\} = E\{x(n)x(n-(i-j))\} = E\{x(n)x(n-l)\} = r_{xy}(l) \]

\[ d_i(n) = E\{x(n-i)y(n)\} = E\{y(n)x(n-i)\} = E\{y(n)x(n-l)\} = r_{xy}(l) \]

\[ c_i(n) = h(n,i) \equiv h(i) = c_i \]

**Discrete Wiener-Hopf Equations for WSS processes**

We form the LMMSE estimate by solving the corresponding set of normal equations for the filter coefficients, \( c_o = h_o \):

\[ Rh_o = d \]

or in matrix form:
\[
\begin{bmatrix}
    r_s(0) & r_s(1) & \cdots & r_s(M-1) \\
    r_s(1) & r_s(0) & \cdots & r_s(M-2) \\
    \vdots & \vdots & \ddots & \vdots \\
    r_s(M-1) & r_s(M-2) & \cdots & r_s(0)
\end{bmatrix}
\begin{bmatrix}
    h_o(0) \\
    h_o(1) \\
    \vdots \\
    h_o(M-1)
\end{bmatrix}
= 
\begin{bmatrix}
    r_y(0) \\
    r_y(1) \\
    \vdots \\
    r_y(M-1)
\end{bmatrix}
\]

Equation 2.6

That is, a time-invariant optimum FIR filter is implemented based upon the convolution:

\[\hat{y}_o(n) = \sum_{k=0}^{M-1} h_o(k)x(n-k)\]

Equation 2.7

where the filter co-efficients satisfy the discrete-time Wiener-Hopf equations:

\[\sum_{k=0}^{M-1} h_o(k)r_s(m-k) = r_{ys}(m) \quad 0 \leq m \leq M-1\]

Equation 2.8

and the MMSE is given by:

\[P_o = P_y - \sum_{k=0}^{M-1} h_o(k)r_{ys}(k) = r_{y}(0) - \sum_{k=0}^{M-1} h_o(k)r_{ys}(k)\]

Equation 2.9

- \(r_s(-l) = r_s(l)\) and \(r_{ys}(-l) = r_{ys}(l)\)
- The correlation matrix, \(R\), for WSS processes is a positive definite, symmetric, Toeplitz matrix.
- A Toeplitz matrix is defined such that \(A = [a_{i,j}] = [a_{-j}]\), \(1 \leq i \leq M, 1 \leq j \leq N\). A square Toeplitz matrix appears as:

\[
A = \begin{bmatrix}
    a_0 & a_{-1} & a_{-2} & \cdots & a_{-N} \\
    a_1 & a_0 & a_{-1} & \cdots & a_{-N} \\
    a_2 & a_1 & a_0 & \cdots & a_{-N} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    a_{N-1} & a_{N-2} & a_{N-3} & \cdots & a_0
\end{bmatrix}
\]

- A symmetric, Toeplitz matrix has \(a_{-i} = a_i\)

Example 2–2

Problem: Consider the harmonic random process:

\[y(n) = A\cos(\omega_o n + \phi)\]

with fixed, but unknown, amplitude and frequency and random uniformly distributed phase. The process is corrupted by additive white Gaussian noise \(v(n) \sim N(0, \sigma_v^2)\) that is uncorrelated with \(y(n)\). The resulting signal \(x(n) = y(n) + v(n)\) is observable. It is required to design an optimum FIR filter with input \(x(n)\) to remove the noise and produce an estimate of the desired response \(y(n)\)

Exercise 2.3: Show that \(r_y(l) = E\{y(n)y(n-l)\} = \frac{1}{2}A^2 \cos \omega_o l\)
2.5 **Linear Prediction**

We are given a set of samples \(x(n), x(n-1), \ldots, x(n-M)\) of a stochastic process of interest and wish to estimate the value of the same process, \(x(n-i)\) (where \(i \not\in [0..M]\)), at a different time \(i\), using a linear combination of the given (known) samples. From [Equation 2.5](#) we have:

\[
\hat{x}(n-i) = -\sum_{k=0}^{M} c_k(n)x(n-k)
\]

where we note that \(\hat{y}(n) \equiv \hat{x}(n-i)\) and we assume the usual convention of negated predictor coefficients (i.e. \(-c_k(n)\)). In terms of the error signal:

\[
e^{i}(n) = x(n-i) - \hat{x}(n-i) = \sum_{k=0}^{M} c_k(n)x(n-k) \quad \text{with } c_j(n) = 1
\]

![Figure 2-3 Illustration showing the samples used in linear signal estimation (Figure 6.16[1])](image)

To determine the MMSE signal estimator we partition as follows:

\[
e^{i}(n) = \sum_{k=0}^{i} c_k(n)x(n-k) + x(n-i) + \sum_{k=i+1}^{M} c_k(n)x(n-k)
\]

\[
= c_1^TX_1(n) + x(n-i) + c_i^TX_2(n)
\]

\[
= [c^{i}(n)]^T\bar{x}(n)
\]

where:

\[
x_1(n) = [x(n) x(n-1) \ldots x(n-(i-1))]^T
\]

\[
x_2(n) = [x(n-(i+1)) x(n-(i+2)) \ldots x(n-M)]^T
\]

\[
c_1(n) = [c_0(n) c_1(n) \ldots c_{i-1}(n)]^T
\]

\[
c_2(n) = [c_{i+1}(n) c_{i+2}(n) \ldots c_M(n)]^T
\]

and:

\[
c^{i}(n) = \begin{bmatrix} c_1(n) \\ c_i(n) \\ c_2(n) \end{bmatrix}, \quad \bar{x}(n) = \begin{bmatrix} x_1(n) \\ x(n-i) \\ x_2(n) \end{bmatrix}
\]

Using the Orthogonality Principle from [Equation 2.4](#) we have that:

\[
E\left[\begin{bmatrix} x_1(n)e^{i}(n) \\ x_2(n)e^{i}(n) \end{bmatrix}\right] = 0
\]

which yields the following set of equations to solve for the predictor co-efficients, \(c_j(n)\):
where for \(j, k = 1, 2\):
\[
\mathbf{R}_{jk}(n) = E\{x_j(n)x_k^T(n)\}
\]
\[
\mathbf{r}_j(n) = E\{x_j(n)x(n-i)\}
\]
and hence \(\mathbf{R}_{21}(n) = \mathbf{R}_{12}^T(n)\).

We also have from the orthogonality principle (see Exercise 2.2) that:
\[
E\{x(n-i)e^{(i)}(n)\} = P_{(i)}^{(i)}(n)
\]
which yields the following:
\[
P_{(i)}^{(i)}(n) = P_{s}(n-i) + \mathbf{r}_1^T(n) \begin{bmatrix} \mathbf{c}_1(n) \\ \mathbf{c}_2(n) \end{bmatrix} = P_{s}(n-i) + \mathbf{r}_1^T(n)\mathbf{c}_1(n) + \mathbf{r}_2^T(n)\mathbf{c}_2(n)
\]
where \(P_{s}(n-i) = E\{|x(n-i)|^2\} = E\{x(n-i)x(n-i)\}\).

**Exercise 2.4** Using the principle of orthogonality derive \(\text{Equation 2.12}\) and \(\text{Equation 2.13}\).

We can combine \(\text{Equation 2.12}\) and \(\text{Equation 2.13}\) into the one set of equations:
\[
\mathbf{R}(n)\mathbf{e}^{(i)}(n) = \begin{bmatrix} 0 \\ P_{(i)}^{(i)}(n) \\ 0 \end{bmatrix} \leftarrow \text{i\textsuperscript{th} row}
\]
\(\text{Equation 2.14}\)

where:
\[
\mathbf{R}(n) = E\{\mathbf{x}(n)\mathbf{x}^T(n)\} = \begin{bmatrix} \mathbf{R}_{11}(n) & \mathbf{r}_1(n) & \mathbf{R}_{12}(n) \\ \mathbf{r}_1^T(n) & P_{s}(n-i) & \mathbf{r}_2^T(n) \\ \mathbf{R}_{12}(n) & \mathbf{r}_2(n) & \mathbf{R}_{22}(n) \end{bmatrix}
\]
\(\text{Equation 2.15}\)

We can use \(\text{Equation 2.14}\) and \(\text{Equation 2.15}\) to form and solve the optimum estimation problem for \(0 \leq i \leq M\) as follows (from Table 6.3[1]).

**Steps for computation of optimal signal estimator \(\hat{x}(n-i)\)**

1. Determine the matrix \(\mathbf{R}(n)\) of the extended data vector \(\mathbf{x}(n)\).
2. Form the \(M \times M\) submatrix \(\mathbf{R}^{(i)}(n)\) of \(\mathbf{R}(n)\) by removing its \(i\text{\textsuperscript{th}}\) row and \(i\text{\textsuperscript{th}}\) column.
3. Form the \(M \times 1\) vector \(\mathbf{d}^{(i)}(n)\) by extracting the \((M + 1) \times 1\) \(i\text{\textsuperscript{th}}\) column, \(\mathbf{d}^{(i)}(n)\), of \(\mathbf{R}(n)\) and removing its \(i\text{\textsuperscript{th}}\) element.
4. Solve the linear system \(\mathbf{R}^{(i)}(n)\mathbf{e}^{(i)}(n) = -\mathbf{d}^{(i)}(n)\) to obtain \(\mathbf{c}^{(i)}(n)\).
5. Compute the MMSE \(P_{(i)}^{(i)}(n) = [\mathbf{d}^{(i)}(n)]^T\mathbf{c}^{(i)}(n)\) where the \((M + 1) \times 1\) vector \(\mathbf{e}^{(i)}(n)\) is \(\mathbf{c}^{(i)}(n)\) with 1 inserted after the \((i-1)\text{\textsuperscript{th}}\) element.
2.5.1 Symmetric Linear Smoother

If \( i = L \) and \( M = 2L \) then we have an \( M \)th order symmetric linear smoother (SLS) that produces an estimate of the middle sample, \( x(n-k) \) for \( k = L \), by using the \( L \) past (\( L+1 \leq k \leq 2L \)) and \( L \) future (\( 0 \leq k \leq L-1 \)) samples. That is:

\[
\hat{x}(n-L) = -\sum_{k=0}^{L-1} c_k(n)x(n-k) - \sum_{k=L+1}^{2L} c_k(n)x(n-k)
\]

and the smoother co-efficients are the solution of Equation 2.12 where:

\[
\begin{align*}
x_1(n) &= \begin{bmatrix} x(n) & x(n-1) & \ldots & x(n-(L-1)) \end{bmatrix}^T \\
x_2(n) &= \begin{bmatrix} x(n-(L+1)) & x(n-(L+2)) & \ldots & x(n-2L) \end{bmatrix}^T \\
c_1(n) &= \begin{bmatrix} c_0(n) & c_1(n) & \ldots & c_{L-1}(n) \end{bmatrix}^T \\
c_2(n) &= \begin{bmatrix} c_{L+1}(n) & c_{L+2}(n) & \ldots & c_{2L}(n) \end{bmatrix}^T
\end{align*}
\]

where \( r_1(n) = E[x_1(n)x(n-L)] \), \( r_2(n) = E[x_2(n)x(n-L)] \), and \( c_k(n) = 1 \)

2.5.2 Forward Linear Prediction

A one-step \( M \)th order \textit{forward linear prediction} (FLP) involves the estimation of the sample, \( x(n) \), by using the \( M \) past samples, \( x(n-1), x(n-2), \ldots, x(n-M) \). From Equation 2.10 this corresponds to the case of \( i = 0 \), thus \( x_1(n) = c_0(n) = 0 \) and \( R_{11}(n) = R_{12}(n) = 0 \), and adopting the following change in notation for the special case of FLP:

\[
\begin{align*}
x(n-1) &\equiv x_2(n) = \begin{bmatrix} x(n-1) & x(n-2) & \ldots & x(n-M) \end{bmatrix}^T \\
a(n) &\equiv \begin{bmatrix} a_1(n) & a_2(n) & \ldots & a_M(n) \end{bmatrix}^T \\
r'^{(f)}(n) &\equiv r_1(n) = E[x(n-1)x(n)] \\
R(n-1) &\equiv R_{22}(n) = E[x_2(n)x_2^T(n)] = E[x(n-1)x(n-1)]
\end{align*}
\]

the predictor co-efficients are the solution of Equation 2.12 that is:

\[
R(n-1)a_{0\cdot}(n) = -r'^{(f)}(n)
\]

and the MMSE power from Equation 2.13 is given by:

\[
P'^{(f)}_o(n) = P_s(n) + r'^{(f)}(n)a_{0\cdot}(n)
\]

where \( P_s(n) = E[x(n)x(n)] \). An estimate of \( x(n) \) is then provided by:

\[
\hat{x}(n) = -\sum_{k=1}^{M} a_k(n)x(n-k)
\]

and the FLP error filter is defined by:

\[
e'^{(f)}(n) = e^{(0)}(n) = x(n) + \sum_{k=1}^{M} a_k(n)x(n-k) = \sum_{k=0}^{M} a_k(n)x(n-k)
\]

- It is standard notational practice to indicate the order of analysis by: \( a_k^{(M)}(n) \equiv a_k(n) \), which identifies the \( k \)th co-efficient of the \( M \)th order FLP.
- Since \( i = 0 \) we have that by definition \( a_{0\cdot}^{(M)}(n) = 1 \)

2.5.3 Backward Linear Prediction

A one-step \( M \)th order \textit{backward linear prediction} (BLP) involves the estimation of the sample, \( x(n-M) \), by using the \( M \) future samples, \( x(n), x(n-1), \ldots, x(n-(M-1)) \). From Equation 2.10
this corresponds to the case of \( i = M \), thus \( x_1(n) = c_2(n) = 0 \) and \( R_{22}(n) = R_{12}(n) = 0 \), and adopting the following change in notation for the special case of BLP:

\[
\begin{align*}
\mathbf{x}(n) &\equiv x_1(n) = [x(n), x(n-1) \ldots x(n-(M-1))]^T \\
\mathbf{b}(n) &\equiv [b_0(n), b_1(n), \ldots, b_{M-1}(n)]^T \equiv [c_0(n), c_1(n), \ldots, c_{M-1}(n)]^T = \mathbf{c}_1(n) \\
r^b(n) &\equiv r_1(n) = E\{\mathbf{x}(n)x(n-M)\} \\
\mathbf{R}(n) &\equiv \mathbf{R}_{11}(n) = E\{\mathbf{x}_1(n)\mathbf{x}_1^T(n)\} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}
\end{align*}
\]

the predictor co-efficients are the solution of Equation 2.12 that is:

\[
\mathbf{R}(n)\mathbf{b}_o(n) = -r^b(n)
\]

and the MMSE power from Equation 2.13 is given by:

\[
P^o_o(n) = P_o(n-M) + r^{bT}(n)\mathbf{b}_o(n)
\]

where \( P_o(n-M) = E\{x(n-M)x(n-M)\} \). An estimate of \( x(n-M) \) is then provided by:

\[
\hat{x}(n-M) = -\sum_{k=0}^{M-1} b_k(n)x(n-k)
\]

and the BLP error filter is defined by:

\[
e^b(n) = e^{(M)}(n) = \sum_{k=0}^{M-1} b_k(n)x(n-k) + x(n-M) = \sum_{k=0}^{M} b_k(n)x(n-k)
\]

- It is standard notational practice to indicate the order of analysis by: \( b^{(M)}_k(n) \equiv b_k(n) \), which identifies the \( k \)th co-efficient of the \( M \)th order BLP.
- Since \( i = M \) we have that by definition \( b^{(M)}_M(n) = 1 \)

### 2.5.4 Stationary Processes

Is the process \( x(n) \) is WSS, then the elements of the correlation matrix \( \mathbf{R}(n) \) and correlation vector \( \mathbf{r}(n) \) no longer depend explicitly on the time-index \( n \) but only on the difference. That is \( r(l) = E\{x(n)x(n-l)\} \) does not depend on \( n \), only on \( l \). Define:

\[
\mathbf{r} = [r(1), r(2), \ldots, r(M)]^T
\]

It is evident that:

\[
\mathbf{r}^f = E\{\mathbf{x}(n-1)x(n)\} = \mathbf{r} \\
\mathbf{r}^b = E\{\mathbf{x}(n)x(n-M)\} = \mathbf{J}\mathbf{r}
\]

where \( \mathbf{J} \) is the exchange matrix:

\[
\mathbf{J} = \begin{bmatrix}
0 & 0 & \cdots & 1 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 1 & \cdots & 0 \\
1 & 0 & \cdots & 0
\end{bmatrix}
\]

which simply reverses the order of the vector elements of \( \mathbf{r} \).

We also note that:

\[
\mathbf{R}(n) = \mathbf{R}(n-1) = \mathbf{R} = \begin{bmatrix}
r(0) & r(1) & \cdots & r(M-1) \\
r(1) & r(0) & \cdots & r(M-2) \\
\vdots & \vdots & \ddots & \vdots \\
r(M-1) & r(M-2) & \cdots & r(0)
\end{bmatrix}
\]

Therefore for the stationary FLP:
and for the stationary BLP:
\[
Rb_o = -Jr
\]
\[
P_o^b = r(0) + r^T Jb_o
\]
Equation 2.17
Noting that \(J^T = J\). Since \(R\) is a symmetric Toeplitz matrix it can be shown that:
\[RJ = JR\]
and hence we have that for WSS signals:
\[b_o = Ja_o\]
\[P_o^f = P_o^b\]
That is the BLP coefficient vector is the reverse of the FLP coefficient vector and the MMSE powers are the identical. In other order words the FLP and BLP co-efficients are duals of one another. This duality property only holds for stationary processes.

**Exercise 2.5** Show that \(RJ = JR\) and hence show that \(b_o = Ja_o\) and \(P_o^f = P_o^b\)

### 2.5.5 Properties

**Property 1**
If the signal \(x(n)\) is stationary, then the symmetric, linear smoother has linear phase.

**Proof:** Using the fact that for stationary signals \(\overline{RJ} = JR\) then from Equation 2.15 we can show that:
\[\overline{c} = J\overline{c}\]
and since the smoother impulse response, \(c\), has even symmetry it has, by definition, linear phase.

**Exercise 2.6** Show that \(\overline{c} = J\overline{c}\) and hence that the smoother has linear phase.

**Property 2**
If the signal \(x(n)\) is stationary, then the FLP error filter \(1, a_1, a_2, \ldots, a_M\) is minimum-phase and the BLP error filter \(b_0, b_1, \ldots, b_{M-1}, 1\) is maximum-phase.

**Proof:** The system function of the FLP error filter \(A(z) = 1 + \sum_{k=1}^{M} a_k z^{-k}\) can be shown to have all zeros inside the unit circle and hence be, by definition, minimum-phase. Since \(b = Ja\), then we have that \(B(z) = z^{-M} A \left( \frac{1}{z} \right)\) which implies that all zeros are outside the unit circle and hence the BLP error filter is maximum-phase.

**Exercise 2.7** Show that \(b = Ja\) implies \(B(z) = z^{-M} A \left( \frac{1}{z} \right)\)
**Example 2–3**

Problem: A random sequence $x(n)$ is generated by passing a white Gaussian noise process $w(n) \sim N(0,1)$ through the filter:

$$x(n) = w(n) + \frac{1}{2} w(n-1)$$

Determine the second-order FLP, BLP and SLS.

### 2.6 Optimum Infinite Impulse Response Filters

For optimum IIR filters the presence of both zeros and poles in the filter transfer function, $H(z)$, implies an infinite impulse response sequence, $h_0(k)$, hence the term IIR filter. The theory for nonstationary signals is complicated and beyond the scope of these notes. For stationary signals, the optimum IIR filter equation, Wiener-Hopf equation, and expression for the MMSE power are the same as [Equation 2.7], [Equation 2.8] and [Equation 2.9] respectively with the exception that the summation depends on whether we are interested in a noncausal or causal IIR filter.

#### Noncausal IIR filter

The noncausal optimum IIR filter is implemented based upon the convolution:

$$\hat{y}_o(n) = \sum_{k=-\infty}^{\infty} h_{nc}(k)x(n-k)$$

where the filter co-efficients satisfy the discrete-time Wiener-Hopf equations:

$$\sum_{k=-\infty}^{\infty} h_{nc}(k)r_x(m-k) = r_{yx}(m) \quad \text{for} \quad -\infty < m < \infty$$

and the MMSE is given by:

$$P_o = r_y(0) - \sum_{k=-\infty}^{\infty} h_{nc}(k)r_{yx}(k)$$

#### Causal IIR filter

The causal optimum IIR filter is implemented based upon the convolution:

$$\hat{y}_o(n) = \sum_{k=0}^{\infty} h_{c}(k)x(n-k)$$

where the filter co-efficients satisfy the discrete-time Wiener-Hopf equations:

$$\sum_{k=0}^{\infty} h_{c}(k)r_x(m-k) = r_{yx}(m) \quad \text{for} \quad 0 \leq m < \infty$$

and the MMSE is given by:

$$P_o = r_y(0) - \sum_{k=0}^{\infty} h_{c}(k)r_{yx}(k)$$

Equation 2.18

Analysis is facilitated if the Wiener-Hopf equations are expressed by the following convolutional equation:
where the complexity of the solution depends on the range of \( m \) that is applicable.

### 2.6.1 Noncausal IIR filter

Since the limits of the summation that apply to \( m \in (-\infty, \infty) \) the convolution property of the \( z \)-transform can be invoked to give:

\[
H_{nc}(z)R_s(z) = R_{yx}(z)
\]

thus:

\[
H_{nc}(z) = \frac{R_{yx}(z)}{R_s(z)}
\]  

where \( H_{nc}(z) \) is the optimum IIR filter transfer function, \( R_s(z) = \sum l r_s(l)z^{-l} \) is the power-spectral density (PSD) of \( x(n) \), and \( R_{yx}(z) = \sum l r_{xy}(l)z^{-l} \) is the cross-PSD between \( y(n) \) and \( x(n) \).

#### Example 2–4

**Problem:** Consider the problem of estimating a desired signal \( y(n) \) that is corrupted by additive noise, \( v(n) \). The goal is to design the optimum IIR filter to extract \( y(n) \) from the noisy observations:

\[
x(n) = y(n) + v(n)
\]

given that \( y(n) \) and \( v(n) \) are uncorrelated signals with known autocorrelation sequences:

\[
r_s(l) = \alpha \delta(l) \quad -1 < \alpha < 1 \quad \text{and} \quad r_{xy}(l) = \sigma_y^2 \delta(l)
\]

respectively where \( \alpha = \frac{4}{5} \) and \( \sigma_y^2 = 1 \).

### 2.6.2 Causal IIR filter

Since the limits of the summation that apply to \( m \in [0, \infty) \) we cannot use the convolution property of the \( z \)-transform to provide an analytic expression for the causal IIR filter. An alternative methodology is to note the following:

1. Any regular process can be transformed to an equivalent white process
2. The solution to the Wiener-Hopf equations for \( m \leq 0 \) is trivial if the input is white

#### Solution for white input processes

If \( x(n) \) is a white noise process then:

\[
r_s(l) = \sigma_s^2 \delta(l)
\]

and from \( \text{Equation 2.19} \) this gives:

\[
h_c(m) * \delta(m) = \frac{r_{xy}(m)}{\sigma_y^2} \quad 0 \leq m < \infty
\]

which implies:
a causal filter response. The optimum IIR filter system function response is:

$$H_\varepsilon(z) = \frac{1}{\sigma_x^2}[R_{ys}(z)]_+$$

Equation 2.25

where

$$[R_{ys}(z)]_+ = \sum_{l=0}^{\infty} r_{ys}(l)z^{-l}$$

is the one-sided $z$-transform of the two-sided sequence $r_{ys}(l)$. From Equation 2.18 and Equation 2.24 the MMSE power is given by:

$$P_e = r_y(0) - \frac{1}{\sigma_x^2} \sum_{k=0}^{\infty} |r_{ys}(k)|^2$$

Equation 2.26

Solution for regular input processes

From stochastic signal theory any regular input process, $x(n)$, can be considered as the output of a linear time-invariant system with transfer function, $H_x(z)$, excited by a white noise source, $w(n)$, with noise variance $\sigma_w^2$. That is:

$$x(n) = \sum_{k=0}^{\infty} h_x(k)w(n-k) \quad \Rightarrow \quad R_x(z) = H_x(z)H_x^\dagger(z^-1)R_w(z) = H_x(z)H_x^\dagger(z^-1)\sigma_w^2$$

For real-valued regular signal, $x(n)$, the PSD can be factored as:

$$R_x(z) = \sigma_x^2H_x(z)H_x(z^-1)$$

where $H_x^\dagger(z) = \frac{1}{H_x(z)}$ is known as the whitening filter since:

$$w(n) = \sum_{k=0}^{\infty} h_x^{-1}(k)x(n-k)$$

That is, by passing $x(n)$ through $H_x^\dagger(z) = \frac{1}{H_x(z)}$ a linearly equivalent white noise process, $w(n)$, is output. Then the optimum causal IIR filter for estimating $y(n)$ from $w(n)$ is provided by Equation 2.25 where $x(n) \equiv w(n)$, that is:

$$H_\varepsilon(z) = \frac{1}{\sigma_x^2}[R_{wy}(z)]_+$$

This is illustrated by Figure 2-4 where it can be seen that the optimum causal IIR filter for estimating $y(n)$ from $x(n)$ is:

$$H_\varepsilon(z) = \left[ \frac{1}{H_x(z)} \right]H_\varepsilon(z)$$
To express \( H_c(z) \) in terms of \( R_{yx}(z) \) a relationship between \( R_{yw}(z) \) and \( R_{yx}(z) \) is needed. From [1, page 298] this relationship is:

\[
R_{yw}(z) = \frac{R_{yx}(z)}{H_x(z^{-1})}
\]

Hence:

\[
H_c(z) = \frac{1}{\sigma^2} \left[ \frac{R_{yx}(z)}{H_x(z^{-1})} \right]
\]

and thus:

\[
H_c(z) = \frac{1}{\sigma^2 H_x(z)} \left[ \frac{R_{yx}(z)}{H_x(z^{-1})} \right]
\]

and the MMSE power, from \textbf{Equation 2.26}, where \( w(n) \) is the linearly equivalent white noise process, can be expressed:

\[
P_c = r_y(0) - \sum_{k=0}^{\infty} h_c(k) r_{yx}(k) = r_y(0) - \frac{1}{\sigma_x^2} \sum_{k=0}^{\infty} |r_{yw}(k)|^2
\]

Since \( R_x(z) = \sigma_x^2 H_x(z) H_x(z^{-1}) \) then from \textbf{Equation 2.20}:

\[
H_{nc}(z) = \frac{R_{yx}(z)}{R_x(z)} = \frac{1}{\sigma^2 H_x(z) H_x(z^{-1})}
\]

and the MMSE power, from \textbf{Equation 2.26}, where \( w(n) \) is the linearly equivalent white noise process, can be expressed:

\[
P_{nc} = r_y(0) - \sum_{k=-\infty}^{\infty} h_{nc}(k) r_{yx}(k) = r_y(0) - \frac{1}{\sigma_x^2} \sum_{k=-\infty}^{\infty} |r_{yw}(k)|^2
\]

- Since \( |r_{yw}(k)| \geq 0 \), then as the order of the filter increases the MMSE decreases due to more \( r_{yw}(k) \) co-efficients contributing to decrease the \( P_y = r_y(0) \) in the expressions for \( P_c \) and \( P_{nc} \).

\textbf{Example 2–5}

The optimum causal IIR filter solution for the problem described in \textbf{Example 2–4} is now derived and compared with the optimum noncausal IIR filter.
2.6.3 Causal IIR linear predictor

The one-step forward IIR linear predictor is a causal IIR optimum filter with desired response \( y(n) \equiv x(n+1) \). The prediction error is:

\[
e^f(n+1) = x(n+1) - \hat{x}(n+1) = x(n+1) - \sum_{k=0}^{\infty} h_{lp}(k)x(n-k)
\]

*Equation 2.29*

To derive the expression for the IIR linear predictor transfer function \( H_{lp}(z) = \sum_{k=0}^{\infty} h_{lp}(k)z^{-k} \)

we note that since \( y(n) = x(n+1) \) then \( r_{xy}(l) = r_{e}(l+1) \) and taking the \( z \)-transform of both sides \( R_{xy}(z) = zR_{e}(z) = z\sigma_x^2 H_x(z)H_x(z^{-1}) \). The optimum linear predictor is:

\[
H_{lp}(z) = \frac{1}{\sigma_x^2 H_x(z)} \left[ \frac{R_{xy}(z)}{H_x(z^{-1})} \right] = \frac{1}{\sigma_x^2 H_x(z)} \left[ \frac{z\sigma_x^2 H_x(z)H_x(z^{-1})}{H_x(z^{-1})} \right] = \left[ \frac{zH_x(z)}{H_x(z)} \right]_1
\]

\[
= \sum_{k=1}^{\infty} h_{k}(k)z^{-k+1}
\]

\[
= \frac{zH_x(z) - h_{x}(0)z}{H_x(z)} = \frac{zH_x(z) - z}{H_x(z)}
\]

where without loss of generality and for convenience it is assumed that \( h_{x}(0) = 1 \).

Of more interest is the prediction error filter transfer function, that is from *Equation 2.29*:

\[
e^f(n) = x(n) - \sum_{k=0}^{\infty} h_{lp}(k)x(n-1-k) \implies H_{pef}(z) = \frac{E^f(z)}{X(z)} = 1 - z^{-1}H_{lp}(z) = - \frac{1}{H_x(z)}
\]

That is, the prediction error filter is identical to the whitening filter of the process and hence the prediction error process is white. It can be shown the MMSE power is given by

\[
P_{o} = P_e = E[|e^f(n)|^2] = \sigma_x^2
\]

which is as expected.

2.7 Application of Optimum Linear Filters

2.7.1 Inverse Filtering and Deconvolution

The inverse filtering or deconvolution problem involves the design of an optimum inverse filter for linearly distorted signals observed in the presence of additive noise. The typical configuration of such a system is shown by Figure 2-5 where \( G(z) \) is the known system response of the linear distortion, \( H(z) \) is the optimum inverse filter we are designing, \( y(n) \) is the desired signal we are trying to recover, \( s(n) \) is the observed (available) linearly distorted signal, \( v(n) \) is additive white noise and \( \hat{y}(n) \) is an estimation of the desired signal.

![Figure 2-5 Optimum Inverse System Modeling (Figure 6.24[1])](image-url)
The delay element is required since the linear distortion filter is causal and its output is delayed by D samples. Usually this is unknown and has to be determined empirically for improved performance. The optimum noncausal IIR filter is derived by:

\[ H_{nc}(z) = \frac{z^{-D}R_{ys}(z)}{R_y(z)} \]

where the \( z^{-D} \) arises because the desired response is \( y(n-D) \). Since \( y(n) \) and \( v(n) \) are uncorrelated, \( x(n) = s(n) + v(n) \) and \( X(z) = S(z) + V(z) = G(z)Y(z) + V(z) \) this gives:

\[ R_{ys}(z) = R_{ys}(z) = G(z^{-1})R_y(z) \]
\[ R_y(z) = G(z)G(z^{-1})R_y(z) + R_v(z) \]

and thus the optimum inverse filter is:

\[ H_{nc}(z) = \frac{z^{-D}G(z^{-1})R_y(z)}{G(z)G(z^{-1})R_y(z) + R_v(z)} \]

which, in the absence of noise, yields the expected result:

\[ H_{nc}(z) = \frac{z^{-D}}{G(z)} \]

If we assume that system is driven by a white noise signal \( y(n) \) with variance \( \sigma_y^2 \) and the additive noise \( v(n) \) is white with variance \( \sigma_v^2 \) then:

\[ H_{nc}(z) = \frac{z^{-D}}{G(z) + [1/G(z^{-1})](\sigma_y^2/\sigma_v^2)} \]

- In most practical cases the system response, \( G(z) \) is unknown and the more difficult problem of blind deconvolution applies.

### 2.7.2 Channel Equalisation in Data Transmission Systems

This important problem is beyond the scope of these notes and students specialising in digital communications are referred to [1, pages 310-319]. Of particular interest is Example 6.8.1 [1, page 317] which discusses the equalisation problem in the context of optimum FIR filtering.

### 2.7.3 Matched Filters and Eigenfilters

An important class of optimum filters are those that maximise the output signal-to-noise ratio. Such filters are used to detect signals in additive noise in many applications, including digital communications and radar. Detailed analysis of this problem is beyond the scope of these notes and interested students should refer to [1, pages 319-325]. However we can illustrate the basic tenets of the problem. Consider the observation vector, \( x(n) \), of a desired signal, \( s(n) \) subject to interfering and/or noise \( v(n) \). That is:

\[ x(n) = s(n) + v(n) \]

The optimum linear filter is designed to produce estimates \( y(n) = \hat{s}(n) \) from the observation vector \( x(n) \). For the optimum FIR filter with co-efficient vector \( c(n) \) the filter output is given by:

\[ y(n) = c^T x(n) = c^T s(n) + c^T v(n) \]
The output signal power is defined as \( P_s(n) = E[|c^T s(n)|^2] = c^T R_s(n)c \) and the output noise power is given by \( P_v(n) = E[|c^T v(n)|^2] = c^T R_v(n)c \). The signal-to-noise ratio for WSS signals as a function of \( c \) is then:

\[
\text{SNR}(c) = \frac{P_s}{P_v} = \frac{c^T R_s c}{c^T R_v c}
\]

Of interest is the special case for deterministic signals, \( s(n) = \alpha s_o \) in additive white noise \((R_v = P_v I)\). It can be shown \([1, \text{page 320}]\) that \( \text{SNR}(c) \) is maximised when \( c_o = \kappa s_o \), that is the filter co-efficients are a scaled replica of the known signal and this type of filter is usually referred to as the matched filter.

### 2.8 References