Linear PEF Properties

1) $A_p(z)$, forward PEF is a Min-Phase filter. The proof is by induction and is in the book.

2) $B_p(z)$, Backward PEF is a Max-phase filter. However, for a perfectly predictable process, all roots of $B_p(z)$ lie on the unit circle. For a perfectly predictable process

$$E_p^f = 0 = \left(1 - |k_m|\right)E_{p-1}^f \Rightarrow |k_m| = 1$$

3) A PEF is a whitening filter as the error becomes a white noise.

4) Orthogonality of the Backward PEF:

$$E\left[g_m(n)g_\ell^*(n)\right] = \begin{cases} 0 & 0 \le \ell \le m - 1 \\ E_m^b & \ell = m \end{cases}$$

The proof is easy, straightforward and in the book. Read the additional properties on page 876.

Levinson-Durbin Algorithm

This algorithm is an efficient method to solve the normal equation: $\sum_{k=0}^{P} a_p(k) \boldsymbol{g}_{xx}(\ell - k) = 0$

$\ell = 1,2,......,P \quad a_p(0) = 1$ In Matrix form: $\boldsymbol{G}_{\underline{xx}} \cdot \underline{a}_p = \underline{0}$ or $\boldsymbol{G}_m a_m = -\boldsymbol{g}_m$ if we take $a_0 = 1$ out

from the vector a_m.

Lets look at $\underline{\boldsymbol{G}}_{xx}$

$$\underline{\Gamma}_{xx}^{\ p} = \begin{bmatrix} \boldsymbol{g}_{xx}(0) & \boldsymbol{g}_{xx}^{*}(1) & \boldsymbol{g}_{xx}^{*}(P-1) \\ \boldsymbol{g}_{xx}(1) & \boldsymbol{g}_{xx}(0) & \boldsymbol{g}_{xx}^{*}(P-2) \\ & & ..... \\ \boldsymbol{g}_{xx}(P-1) & \boldsymbol{g}_{xx}(P-2) & \boldsymbol{g}_{xx}(0) \end{bmatrix}$$

Since $\boldsymbol{G}(i,j) = \boldsymbol{G}(i - j) \rightarrow \boldsymbol{G}_{xx}$ is a Teoplitz matrix. Also, $\boldsymbol{G}(i,j) = \boldsymbol{G}^{*}(j, i)$ is a Hermitian matrix.

Then, the matrix can be rewritten as:

$$\underline{\Gamma}_{xx}(m) = \begin{bmatrix} \Gamma_{m-1} & \boldsymbol{g}_{m-1}^{b*} \\ \boldsymbol{g}_{m-1}^{bt} & \boldsymbol{g}_{xx}(0) \end{bmatrix} \text{ where } \underline{\boldsymbol{g}}_{m-1}^{b} = \begin{bmatrix} \boldsymbol{g}(m-1) \\ \boldsymbol{g}(m-2) \\ ... \\ \boldsymbol{g}(1) \end{bmatrix} \quad \underline{\boldsymbol{g}}_{m-1}^{bT} = \left[\underline{\boldsymbol{g}}_{m-1}^{b} \right]^{T}$$

Then lets write

$$\underline{a}_m = \begin{bmatrix} a_m(1) \\ ... \\ ... \\ a_m(m) \end{bmatrix} = \begin{bmatrix} a_{m-1}(1) \\ a_{m-1}(2) \\ ... \\ 0 \end{bmatrix} + \begin{bmatrix} d_{m-1} \\ ... \\ ... \\ k_m \end{bmatrix}$$

Then the normal equation $\underline{\Gamma}_m \cdot \underline{a}_m = -\underline{\boldsymbol{g}}_m$ can be written as

$$\begin{bmatrix} \underline{\Gamma}_{m-1} & \underline{\boldsymbol{g}}_{m-1}^{b*} \\ \underline{\boldsymbol{g}}_{m-1}^{bt} & \boldsymbol{g}(0) \end{bmatrix} \left\{ \begin{bmatrix} a_{m-1} \\ 0 \end{bmatrix} + \begin{bmatrix} d_{m-1} \\ k_m \end{bmatrix} \right\} = - \begin{bmatrix} \boldsymbol{g}_{m-1} \\ \boldsymbol{g}_{(m)} \end{bmatrix}$$

This leads to recursive Levinson-Durbin Equations:

Results:

1) $a_m(m) = k_m = -\dfrac{\boldsymbol{g}_{xx}(m) + \underline{\boldsymbol{g}}_{m-1}^{bt} \underline{a}_{m-1}}{\boldsymbol{g}_{xx}(0) + \underline{\boldsymbol{g}}_{m-1}^{bt} \underline{a}_{m-1}^{b*}} = -\dfrac{\boldsymbol{g}_{xx}(m) + \underline{\boldsymbol{g}}_{m-1}^{bt} \underline{a}_{m-1}}{E_{m-1}^{f}}$

2) also $E_m^{f} = E_{m-1}^{f}\left(1 - |k_m|^2\right) \quad m = 1,2,.....P$

3) $a_m(k) = a_{m-1}(k) + k_m a_{m-1}^{*}(m - k) \quad k = 1,2,...m-1 \quad m = 1,2,....P$

Note: The equation (1) is written wrong in the text. Eq. 11.3.16, page 867 the denominator should be E_{m-1}^f . As an example, let $P = 3$. The calculation starts at $m = 1, 2, 3$. So, we have:

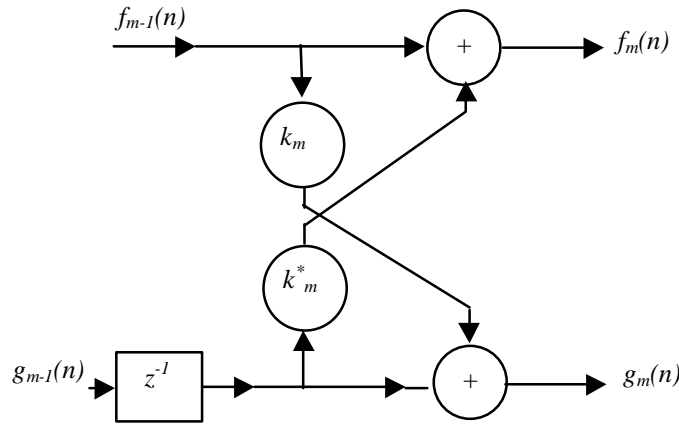Stage 1: PEF of $m = 1$ Stage 2: $m = 2$ Stage 3 $m = 3$

$$\begin{cases} a_1(0) = 1 \\ a_1(1) = k_1 = \dfrac{g(1)}{g(0)} \\ E_1 = E_0\left(1 - |k_1|^2\right) \\ E_0 = g_{xx}(0) = s_x^2 \end{cases}$$

$$\begin{cases} a_2(0) = 1 \\ a_2(1) = k_1 + k_2 k_1^* \\ a_2(2) = k_2 \\ E_2 = E_1(1 - |k_1|^2) \end{cases}$$

$$\begin{cases} a_3(0) = 1 \\ a_3(1) = a_2(1) + k_3 k_2^* \\ a_3(2) = k_2 + k_3 a_2^*(1) \\ a_3(3) = k_3 \\ E_3 = E_2(1 - |k_3|^2) \end{cases}$$

"Burg Algorithm" page 926

Each stage is being optimized due to the orthogonality of backward prediction error.

At each stage: $\begin{cases} f_m(n) = f_{m-1}(n) + k_m^* g_{m-1}(n-1) \\ g_m(n) = g_{m-1}(n-1) + k_m f_{m-1}(n) \end{cases}$, $m = 1, 2, ..., P$

The cost function to be minimized is

$$e_m = E\left\{ |f_m(n)|^2 + |g_m(n)|^2 \right\}$$

subject to the constraint that $a_m(k)$ satisfy Levinson-Durbin recursion. Let $k_m = a_m + jb_m$ at each stage. Then taking gradient of e_m with respect to a_m and b_m and make it equal to zero, we get:

$$(*) \quad k_{mo} = -\frac{2E\left\{ g_{m-1}(n-1) \cdot f_{m-1}^*(n) \right\}}{E\left\{ |f_{m-1}(n)|^2 + |g_{m-1}(n-1)|^2 \right\}} \qquad m = 1, 2, ...P$$

optimum

Assuming that $x(n)$ is ergodic, we can replace ensemble averaging (expected value) by sample (time) averaging. Then we will get:

$$\hat{k}_{mo} = -\frac{2 \sum\limits_{n=m+1}^{N} g_{m-1}(n-1) \cdot f_{m-1}^{*}(n)}{\underbrace{\sum\limits_{n=m+1}^{N} \left[\left| f_{m-1}(n) \right|^2 + \left| g_{m-1}(n-1) \right|^2 \right]}_{\hat{E}_{m-1}}}$$

Also, $\hat{E}_{m} = \left(1 - \left| k_{mo} \right|^2\right)\hat{E}_{m-1}$. This equation is written wrong in the text Eq. 12.3.19 page 927.

Proof of the above equation:

We prove it for E_{m}^{f}, the rest will follow.

$$f_{m}(n) = f_{m-1}(n) + k_{mo}^{*} g_{m-1}(n-1)$$

$$E\left\{\left| f_{m}(n) \right|^2\right\} = E\left\{\left| f_{m-1}(n) \right|^2\right\} + \left| k_{mo} \right|^2 E\left\{\left| g_{m-1}(n-1) \right|^2\right\}$$
$$+ k_{mo}^{*} E\left\{f_{m-1}^{*}(n) g_{m-1}(n-1)\right\}$$
$$+ k_{mo} E\left\{f_{m-1}(n) g_{m-1}^{*}(n-1)\right\}$$

Now use the equation for k_{mo} from Eq (*) of last page:

$$E\left\{\left| f_{m}(n) \right|^2\right\} = E\left\{\left| f_{m-1}(n) \right|^2\right\} + \left| k_{mo} \right|^2 E\left\{\left| g_{m-1}(n-1) \right|^2\right\}$$
$$+ k_{mo}^{*} \left[\frac{-1}{2} k_{mo} \left[E\left\{\left| f_{m-1}(n) \right|^2\right\} + E\left\{\left| g_{m-1}(n-1) \right|^2\right\}\right]\right]$$
$$+ k_{mo} \left[\frac{-1}{2} k_{mo}^{*} \left[E\left\{\left| f_{m-1}(n) \right|^2\right\} + E\left\{\left| g_{m-1}(n-1) \right|^2\right\}\right]\right]$$
$$= E\left\{\left| f_{m-1}(n) \right|^2\right\} - \left| k_{mo} \right|^2 E\left\{\left| f_{m-1}(n) \right|^2\right\}$$
$$= \left(1 - \left| k_{mo} \right|^2\right) E\left\{\left| f_{m-1}(n) \right|^2\right\}$$

$\Rightarrow E_{m}^{f} = \left(1 - \left| k_{mo} \right|^2\right) E_{m-1}^{f}$ proof is complete.

Therefore, the difference between Burg and Levinson-Durbin algorithms are in the calculation of k_m. Other than that, in both $a_m(k) = a_{m-1}(k) + a_{m-1}^{*}(m-k)$ $\{m = 1,...,P; \quad k = 1,..., m-1\}$ and $E_m = (1 - |k_m|^2) E_{m-1}$

Selection of the AR Model Order – Section 12.3.6

The most common criteria that selects the order are:

1) Akaike Information Criterion (AIC) (1979) that is based on selecting the order that minimizes

$$AIC(P) = \ell n\, \boldsymbol{s}_{wp}^2 + \frac{2P}{N} \quad \text{where } \boldsymbol{s}_{wp}^2 = E_p = MSE$$

2) MDL (Minimization of the description Length) proposed by Rissanen (1983):

$$MDL(P) = N\ell n\,(\boldsymbol{s}_{wp}^2) + P\ell n(N)$$

Example 1:

Lets consider a perfectly predictable signal: $S_n = Ae^{j\boldsymbol{f}}\,e^{j\boldsymbol{w}\,on}$, where $\boldsymbol{f}$ is a random variable.

$$\boldsymbol{g}_{xx}(\ell) = E\left[S_n^* S_{n+\ell}\right] = E\left[A^2 e^{-j\boldsymbol{f}} e^{-j\boldsymbol{w}_o n} e^{j\boldsymbol{w}_o(n+\ell)} e^{j\boldsymbol{f}}\right] = A^2 e^{j\boldsymbol{w}_o\ell}$$

Normal Equation for PEF, $P = 1$:

$$\begin{bmatrix} A^2 & A^2 e^{-j\boldsymbol{w}_o} \\ A^2 e^{j\boldsymbol{w}_o} & A^2 \end{bmatrix}\begin{bmatrix} 1 \\ a_1(1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \because \boldsymbol{s}_w^2 = 0 \text{ no noise!}$$

$$\Rightarrow A^2\left(1 + e^{-j\boldsymbol{w}_o} a_1(1)\right) = 0 \Rightarrow a_1(1) = -e^{+j\boldsymbol{w}_o} \quad \text{PEF } A_1(z) = 1 - e^{-j\boldsymbol{w}o} z^{-1}$$

It has one zero on the unit circle. It is a perfectly predictable signal and prediction depends on frequency $\boldsymbol{w}_0$, not A and not $\boldsymbol{f}$.

What happens if we go for higher order? Let $P = 2$ then,

$$A^2 = \begin{bmatrix} 1 & e^{-j\boldsymbol{w}_o} & e^{-j2\boldsymbol{w}_o} \\ e^{j\boldsymbol{w}_o} & 1 & e^{-j\boldsymbol{w}_o} \\ e^{+j2\boldsymbol{w}_o} & e^{j\boldsymbol{w}_o} & 1 \end{bmatrix}\begin{bmatrix} 1 \\ a_2(1) \\ a_2(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

This reduces to:

$$\begin{bmatrix} 1 & e^{-j\boldsymbol{w}_o} \\ e^{j\boldsymbol{w}_o} & 1 \end{bmatrix}\begin{bmatrix} a_2(1) \\ a_2(2) \end{bmatrix} = -\begin{bmatrix} e^{-j\boldsymbol{w}_o} \\ e^{j2\boldsymbol{w}_o} \end{bmatrix}$$

but this is a singular matrix. We need noise so that the matrix equation could be solved.

Example 2:

Let $x(n) = Ae^{j\boldsymbol{f}} e^{j\boldsymbol{w}_o n} + w(n)$ where $w(n)$ is a white noise.

$$\boldsymbol{g}_{xx}(\ell) = A^2 e^{j\boldsymbol{w}_o\ell} + \boldsymbol{g}_{ww}(\ell) \text{ and } \boldsymbol{g}_{ww}(\ell) = \begin{cases} \boldsymbol{s}_w^2 & \ell = o \\ o & else \end{cases}$$

Note that $\boldsymbol{g}_{sw} = \boldsymbol{g}_{ws} = 0 \because$ signal and noise are uncorrelated here.

Also from previous equation we have: $g_{xx}(0) = A^2 + s_w^2$.

Lets normalize j_{xx} as $g_N(\ell) = \dfrac{g_{xx}(\ell)}{g_{xx}(0)} = \dfrac{A^2 e^{jw_o \ell}}{A^2 + s_w^2}$ $\ell \neq o$

Let l be defined as: $l = \dfrac{A^2}{A^2 + s_w^2} = \dfrac{A^2/s_w^2}{1 + A^2/s_w^3} = \dfrac{SNR}{SNR+1}$ Clearly $o \leq l \leq 1$

$\rightarrow g_N(\ell) = l\, e^{jw_o \ell}$ for $\ell \neq 0$ and $g_N(\ell) = 1$ for $\ell = 0$.

Now the first order PEF:

$$\begin{bmatrix} 1 & le^{-jw_o} \\ le^{jw_o} & 1 \end{bmatrix}\begin{bmatrix} 1 \\ a_1(1) \end{bmatrix} = \begin{bmatrix} s_1^2 \\ 0 \end{bmatrix} \quad s_1^2 = E_1 = M.MSE \text{ at order 1.}$$

$$\Rightarrow \begin{cases} 1 + a_1(1)le^{-jw_o} = s_1^2 \\ le^{jw_o} + a_1(1) = 0 \end{cases} \Rightarrow \begin{cases} a_1(1) = k_1 = -le^{+jw_o} \\ s_1^2 = 1 - l^2 \end{cases}$$

2nd Order PEF:

$$\begin{bmatrix} 1 & le^{-jw_o} & le^{-j2w_o} \\ le^{jw_o} & 1 & le^{-jw_o} \\ le^{j2w_o} & le^{jw_o} & 1 \end{bmatrix}\begin{bmatrix} 1 \\ a_2(1) \\ a_2(2) \end{bmatrix} = \begin{bmatrix} s_2^2 \\ 0 \\ 0 \end{bmatrix}$$

$\Rightarrow a_2(1) = \dfrac{-l}{1+l} e^{jw_o}$ and $a_2(2) = k_2 = \dfrac{-l}{1+l} e^{2jw_o}$

$\Rightarrow A_2(z) = 1 - \dfrac{l}{1+l} e^{jw_o} z^{-1} - \dfrac{l}{1+l} e^{j2w_o} z^{-2} \quad s_2^2 = s_1^2\left(1 - \left|\dfrac{-l}{1+l}\right|^2\right)$

If $SNR = 0\,dB \circledR SNR = 1 \rightarrow l = \dfrac{1}{2}$

$\Rightarrow A_1(z) = 1 - \dfrac{1}{2} e^{jw_o} z^{-1}$ and $z_1 = \dfrac{1}{2} e^{jw_o}$

$$A_2(z) = -1 - \dfrac{1}{3} e^{jw_o} z^{-1} - \dfrac{1}{3} e^{j2w_o} z^{-2} \Rightarrow z_{1,2} = e^{jw_o}\left(\dfrac{1 \pm \sqrt{13}}{6}\right)$$

Note that as the order increases, the roots get closer to the unit circle.

$MMSE_1 = s_1^2 = \dfrac{3}{4}$ and $MMSE_2 = s_2^2 = \dfrac{2}{3}$

As P increases, *MSE* gets closer to $\mathbf{s}_w^2$. As a rule of thumb $P < \dfrac{N}{4}$, where N is the length of the data. Also note that the prediction methods apply only to *WSS* signals.