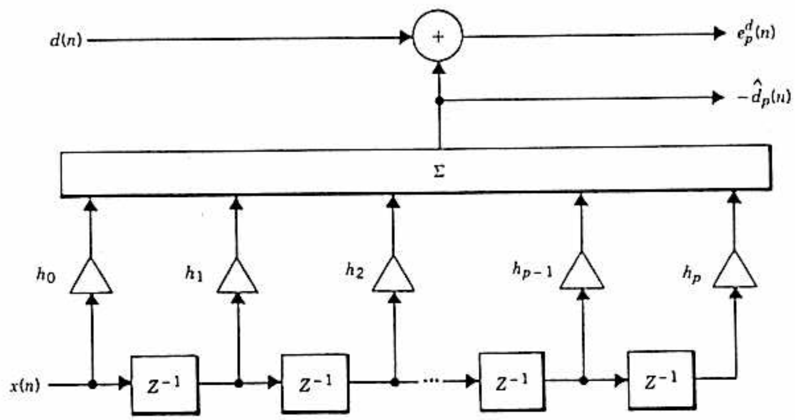


**Figure 11.1** Direct-form realization of the joint process structure.

## 11.1 THE STOCHASTIC WIENER FILTERING AND DETERMINISTIC LEAST SQUARES PROBLEMS

The problem of interest is described in Figure 11.1. The goal is to filter the time series $x(n)$ with a causal, finite impulse response (FIR) digital filter to yield an estimate of the time series $d(n)$. The error in that estimate is denoted by $e^d(n)$

$$e^d(n) = d(n) + \sum_{k=0}^{p} h_k x(n-k) \tag{11.1}$$

Written in matrix form, Eq. 11.1 becomes

$$e^d(n) = d(n) + \begin{bmatrix} h_0 & h_1 & \cdots & h_p \end{bmatrix} \begin{bmatrix} x(n) \\ x(n-1) \\ \vdots \\ x(n-p) \end{bmatrix} \tag{11.2}$$

or, in vector notation,

$$e^d(n) = d(n) + \mathbf{h}^T \mathbf{x} \tag{11.3}$$

The solution to this problem for the unknown filter vector $\mathbf{h}$ can be approached in two ways, depending on the optimality criterion chosen

$$\underset{\mathbf{h}}{\text{minimize}}\ E\left[|e^d(n)|^2\right] \quad \text{Wiener filtering problem} \tag{11.4}$$

$$\underset{\mathbf{h}}{\text{minimize}} \sum_n |e^d(n)|^2 \quad \text{Least squares problem} \tag{11.5}$$



The Wiener filtering problem takes a stochastic viewpoint where the filter is optimized in an expected value sense (minimum mean-squared error). As an alternative, the least-squares problem views the time series involved as deterministic and the filter is optimized for the specific sequences of record (minimum sum-squared error). Both viewpoints give rise to similar mathematical expressions for $\mathbf{h}$.

The distinction between the Wiener filtering and least squares problems is made in the definitions of $\phi_{00}^d$, $\mathbf{g}$, and $\Phi$ where

$$\mathbf{g} = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_p \end{bmatrix} \tag{11.6}$$

$$\Phi = \begin{bmatrix} \phi_{00} & \cdots & \phi_{0p} \\ \vdots & & \vdots \\ \phi_{p0} & \cdots & \phi_{pp} \end{bmatrix} = \begin{bmatrix} \phi_{00} & \cdots & \phi_{0p} \\ \vdots & & \vdots \\ \phi_{0p}^* & \cdots & \phi_{pp} \end{bmatrix} \tag{11.7}$$

For the Wiener filtering problem

$$\phi_{00}^d = E[d(n)\, d(n)^*] \tag{11.8}$$

$$\mathbf{g} = \{E[d(n)x(n-j)^*]\} = \{g_j\} \tag{11.9}$$

$$\Phi = \{E[x(n-k)x(n-j)^*]\} = \{\phi_{kj}\} \tag{11.10}$$

And, for the least squares problem

$$\phi_{00}^d = \sum_n d(n)\, d(n)^* \tag{11.11}$$

$$\mathbf{g} = \left\{\sum_n d(n)x(n-j)^*\right\} = \{g_j\} \tag{11.12}$$

$$\Phi = \left\{\sum_n x(n-k)x(n-j)^*\right\} = \{\phi_{kj}\} \tag{11.13}$$

The next step in solving for the unknown filter vector $\mathbf{h}$ is to write an expression for squared error based on Eq. 11.3

$$|e^d(n)|^2 = d(n)^* d(n) + d(n)^* \mathbf{h}^T \mathbf{x} + \mathbf{h}^H \mathbf{x}^* d(n) + \mathbf{h}^H \mathbf{x}^* \mathbf{h}^T \mathbf{x} \tag{11.14}$$

By making use of the definitions of $\phi_{00}^d$, $\mathbf{g}$, and $\Phi$ the average squared error $E_p^d(\mathbf{h})$ can be derived from Eq. 11.14 for both the Wiener filtering and least squares problems as

$$E_p^d(\mathbf{h}) = \phi_{00}^d + \mathbf{g}^H \mathbf{h} + \mathbf{h}^H \mathbf{g} + \mathbf{h}^H \Phi^T \mathbf{h} \tag{11.15}$$

Now, minimizing $E_p^d(\mathbf{h})$ with respect to $\mathbf{h}$ leads to the following equation.

$$0 = \mathbf{g} + \Phi^T\mathbf{h} \tag{11.16}$$

or

$$\Phi^T\mathbf{h} = -\mathbf{g} \tag{11.17}$$

Expressing Eq. 11.17 in expanded form will be of interest in the next section

$$\begin{bmatrix} \phi_{00} & \cdots & \phi_{0p}^* \\ \vdots & & \vdots \\ \phi_{0p} & \cdots & \phi_{pp} \end{bmatrix} \begin{bmatrix} h_0 \\ \vdots \\ h_p \end{bmatrix} = - \begin{bmatrix} g_0 \\ \vdots \\ g_p \end{bmatrix} \tag{11.18}$$

or

$$\sum_{k=0}^{p} \phi_{kj} h_k = -g_j \qquad j = 0, 1, \ldots, p \tag{11.19}$$

Now, solving Eq. 11.17 for $\mathbf{h}$ yields

$$\mathbf{h} = -(\Phi^T)^{-1}\mathbf{g} \tag{11.20}$$

By substituting the solution for $\mathbf{h}$ back into Eq. 11.15, the minimum value of average squared error can be obtained as

$$E_p^d = \phi_{00}^d + \mathbf{g}^H\mathbf{h} \tag{11.21}$$

Expressing 11.21 in expanded form also will be of interest in the next section.

$$E_p^d = \phi_{00}^d + [g_0^* \ \cdots \ g_p^*] \begin{bmatrix} h_0 \\ \vdots \\ h_p \end{bmatrix} \tag{11.22}$$

or

$$E_p^d = \phi_{00}^d + \sum_{k=0}^{p} g_k^* h_k \tag{11.23}$$

An important property of the solution for $\mathbf{h}$ in Eq. 11.20 is known as the orthogonality principle. From Eq. 11.3, the product of the error $e^d(n)$ and the complex conjugate of the data available to the filter $\mathbf{x}^*$ can be written as

$$\begin{aligned} e^d(n)\mathbf{x}^* &= d(n)\mathbf{x}^* + \mathbf{h}^T\mathbf{x}\mathbf{x}^* \\ &= d(n)\mathbf{x}^* + \mathbf{x}^*\mathbf{x}^T\mathbf{h} \end{aligned} \tag{11.24}$$



Making use of the definitions of $\mathbf{g}$ and $\Phi$ in Eq. 11.6 through Eq. 11.13 and the expression for $\mathbf{h}$ in Eq. 11.17, the average value of Eq. 11.24 is

$$\mathbf{g} + \Phi^T\mathbf{h} = 0 \tag{11.25}$$

Thus, when the filter vector $\mathbf{h}$ is selected so that the average squared error is minimized, the error is orthogonal to the data.

The expressions of $\Phi$ in Eq. 11.10 and Eq. 11.13 have been left general. When the time series in question are stationary (Wiener filtering problem) or have been windowed (least squares problem), $\Phi$ will be Toeplitz

$$\phi_{kj} = \phi_{j-k} \tag{11.26}$$

Thus, all entries along a given diagonal of $\Phi$ are equal. Advantage can be taken of this special structure in the solution of Eq. 11.7, which avoids the necessity of directly inverting $\Phi$ as is done in Eq. 11.20.

## 11.2 BLOCK PROCESSING APPROACH

A common approach to the processing of time series is to split the observation interval into subintervals (perhaps overlapped by some factor). The collection of samples in any given subinterval then are processed as a block. Here we will assume that over the duration of a block, the time series in question are stationary (Wiener filtering problem) or have been windowed (least squares problem). The Toeplitz structure of $\Phi$ will be taken advantage of in solving for the filter. As will be seen, the general filter solution $\{h_0, h_1, \ldots, h_p\}$ has the solution to the linear prediction problem embedded in it.

### 11.2.1 Linear Prediction

One-step forward and backward linear predictors of order $p$ are shown in Figure 11.2. The one-step forward linear predictor weights the most recent $p$ samples $\{x(n-1), \ldots, x(n-p)\}$ of the time series with coefficients $\{a_1, \ldots, a_p\}$ to form an estimate of $-x(n)$. Similarly, the one-step backward linear predictor weights the current and most recent $p-1$ samples $\{x(n), \ldots, x(n-p+1)\}$ of the time series with coefficients $\{b_0, \ldots, b_{p-1}\}$ to form an estimate of $-x(n-p)$. Note that linear predictors are simply special cases of the general filtering problem shown in Figure 11.1 where $x(n)$ and $d(n)$ are drawn from the same underlying time series.

Written explicitly in matrix form, the equation that must be solved to determine the forward linear predictor coefficients is

$$\begin{bmatrix} \phi_0 & \cdots & \phi_{p-1}^* \\ \vdots & & \vdots \\ \phi_{p-1} & \cdots & \phi_0 \end{bmatrix} \begin{bmatrix} a_1^p \\ \vdots \\ a_p^p \end{bmatrix} = - \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_p \end{bmatrix} \tag{11.27}$$

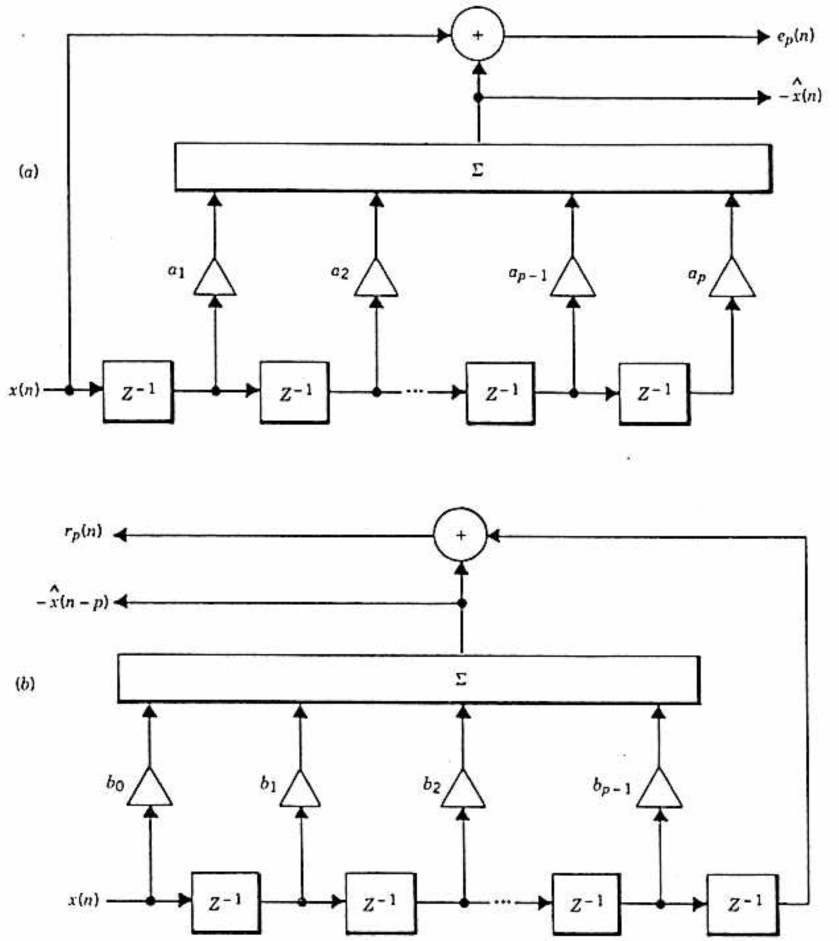


**Figure 11.2** *(a)* One-step forward prediction error filter. *(b)* One-step backward prediction error filter.

Adding $[\phi_1 \ldots \phi_p]^T$ to both sides of Eq. 11.27 yields

$$\begin{bmatrix} \phi_1 & \phi_0 & \cdots & \phi_{p-1}^* \\ \vdots & \vdots & & \vdots \\ \phi_p & \phi_{p-1} & \cdots & \phi_0 \end{bmatrix} \begin{bmatrix} 1 \\ a_1^p \\ \vdots \\ a_p^p \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.28}$$

Now, augmenting Eq. 11.28 with the expression for minimum forward prediction error



$$E_p^\epsilon = \phi_{00} + \sum_{k=1}^{p} \phi_k^* a_k^p \tag{11.29}$$

yields

$$\begin{bmatrix} \phi_0 & & \cdots & \phi_p^* \\ \phi_1 & \phi_0 & \cdots & \phi_{p-1}^* \\ \vdots & \vdots & & \vdots \\ \phi_p & \phi_{p-1} & \cdots & \phi_0 \end{bmatrix} \begin{bmatrix} 1 \\ a_1^p \\ \vdots \\ a_p^p \end{bmatrix} = \begin{bmatrix} E_p^\epsilon \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.30}$$

Thus, the solution to Eq. 11.30 provides both the $p$ forward prediction error filter coefficients $\{a_1^p, \ldots, a_p^p\}$ as well as the minimum error $E_p^\epsilon$. Following these same steps for the backward prediction error filter yields

$$\begin{bmatrix} \phi_0 & \cdots & \phi_{p-1}^* & \phi_p^* \\ \vdots & & \vdots & \vdots \\ \phi_{p-1} & \cdots & \phi_0 & \phi_1^* \\ \phi_p & \cdots & & \phi_0 \end{bmatrix} \begin{bmatrix} b_0^p \\ \vdots \\ b_{p-1}^p \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ E_p^r \end{bmatrix} \tag{11.31}$$

The solution to Eq. 11.31 provides both the $p$ backward prediction error filter coefficients $\{b_0^p, \ldots, b_{p-1}^p\}$ as well as the minimum error $E_p^r$. Although the backward prediction error filter will be carried along explicitly, since $\mathbf{\Phi}$ is Toeplitz

$$\begin{bmatrix} b_0^p \\ \vdots \\ b_{p-1}^p \\ 1 \end{bmatrix} = \begin{bmatrix} a_p^{p*} \\ \vdots \\ a_1^{p*} \\ 1 \end{bmatrix} \tag{11.32}$$

and

$$E_p^r = E_p^\epsilon \tag{11.33}$$

Given that the forward and backward prediction error filters of order $p$ have been determined, it now is of interest to derive order update recursions that will permit generating the prediction error filters of order $p+1$ from those of order $p$. Starting with the forward prediction error filter, simply guess that the correct filter of order $p+1$ is $[1 \ a_1^p \ldots a_p^p \ 0]^T$. Now Eq. 11.30 becomes

$$\begin{bmatrix} \phi_0 & \cdots & \phi_p^* & \phi_{p+1}^* \\ \vdots & & \vdots & \vdots \\ \phi_p & \cdots & \phi_0 & \phi_1^* \\ \phi_{p+1} & \cdots & \phi_1 & \phi_0 \end{bmatrix} \begin{bmatrix} 1 \\ a_1^p \\ \vdots \\ a_p^p \\ 0 \end{bmatrix} = \begin{bmatrix} E_p^\epsilon \\ 0 \\ \vdots \\ 0 \\ \Delta_{p+1}^\epsilon \end{bmatrix} \tag{11.34}$$

where

$$\Delta^e_{p+1} = \phi_{p+1} + \phi_p a^p_1 + \cdots + \phi_1 a^p_p \tag{11.35}$$

Making a similar guess for the backward prediction error filter of order $p + 1$ yields

$$\begin{bmatrix} \phi_0 & \phi_1^* & \cdots & \phi_{p+1}^* \\ \phi_1 & \phi_0 & \cdots & \phi_p^* \\ \vdots & \vdots & & \vdots \\ \phi_{p+1} & \phi_p & \cdots & \phi_0 \end{bmatrix} \begin{bmatrix} 0 \\ b^p_0 \\ \vdots \\ b^p_{p-1} \\ 1 \end{bmatrix} = \begin{bmatrix} \Delta^r_{p+1} \\ 0 \\ \vdots \\ 0 \\ E^r_p \end{bmatrix} \tag{11.36}$$

where

$$\Delta^r_{p+1} = \phi_1^* b^p_0 + \cdots + \phi_p^* b^p_{p-1} + \phi_{p+1}^* \tag{11.37}$$

Notice that if the discrepancy terms $\Delta^e_{p+1}$ and $\Delta^r_{p-1}$ were 0, the proposed forward and backward prediction error filters of order $p + 1$ would be correct. Since the discrepancy terms generally are not 0, the next step involves modifying the proposed filter coefficients so as to cancel the discrepancy. The modification involves adding a weighted version of the proposed backward prediction error filter vector to the proposed forward prediction error filter vector and vice versa

$$\begin{bmatrix} 1 \\ a^{p+1}_1 \\ \vdots \\ a^{p+1}_p \\ a^{p+1}_{p+1} \end{bmatrix} = \begin{bmatrix} 1 \\ a^p_1 \\ \vdots \\ a^p_p \\ 0 \end{bmatrix} + K^r_{p+1} \begin{bmatrix} 0 \\ b^p_0 \\ \vdots \\ b^p_{p-1} \\ 1 \end{bmatrix} \tag{11.38}$$

yielding

$$\begin{bmatrix} E^e_{p+1} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} E^e_p \\ 0 \\ \vdots \\ 0 \\ \Delta^e_{p+1} \end{bmatrix} + K^r_{p+1} \begin{bmatrix} \Delta^r_{p+1} \\ 0 \\ \vdots \\ 0 \\ E^r_p \end{bmatrix} \tag{11.39}$$

and

$$\begin{bmatrix} b^{p+1}_0 \\ b^{p+1}_1 \\ \vdots \\ b^{p+1}_p \\ 1 \end{bmatrix} = K^e_{p+1} \begin{bmatrix} 1 \\ a^p_1 \\ \vdots \\ a^p_p \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ b^p_0 \\ \vdots \\ b^p_{p-1} \\ 1 \end{bmatrix} \tag{11.40}$$



yielding

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ E^r_{p+1} \end{bmatrix} = K^e_{p+1} \begin{bmatrix} E^e_p \\ 0 \\ \vdots \\ 0 \\ \Delta^e_{p+1} \end{bmatrix} + \begin{bmatrix} \Delta^r_{p+1} \\ 0 \\ \vdots \\ 0 \\ E^r_p \end{bmatrix} \tag{11.41}$$

In order to cancel the discrepancy terms

$$K^r_{p+1} = -\frac{\Delta^e_{p+1}}{E^r_p} \tag{11.42}$$

and

$$K^e_{p+1} = -\frac{\Delta^r_{p+1}}{E^e_p} \tag{11.43}$$

Thus, the weighting factors each consist of the discrepancy normalized by the prediction error from the filter of order $p$. Since $\Phi$ is Toeplitz,

$$\Delta^r_{p+1} = \Delta^{e*}_{p+1} \tag{11.44}$$

and

$$K^e_{p+1} = K^{r*}_{p+1} \tag{11.45}$$

Now, by substituting Eqs. 11.42 and 11.43 into the update equations for the forward and backward prediction errors,

$$E^e_{p+1} = E^e_p(1 - K^r_{p+1}K^e_{p+1}) \tag{11.46}$$

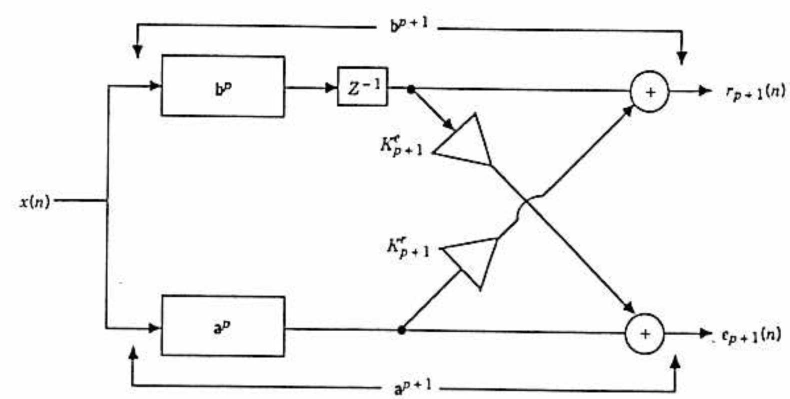


**Figure 11.3** Order update procedure for the forward and backward prediction error filters.

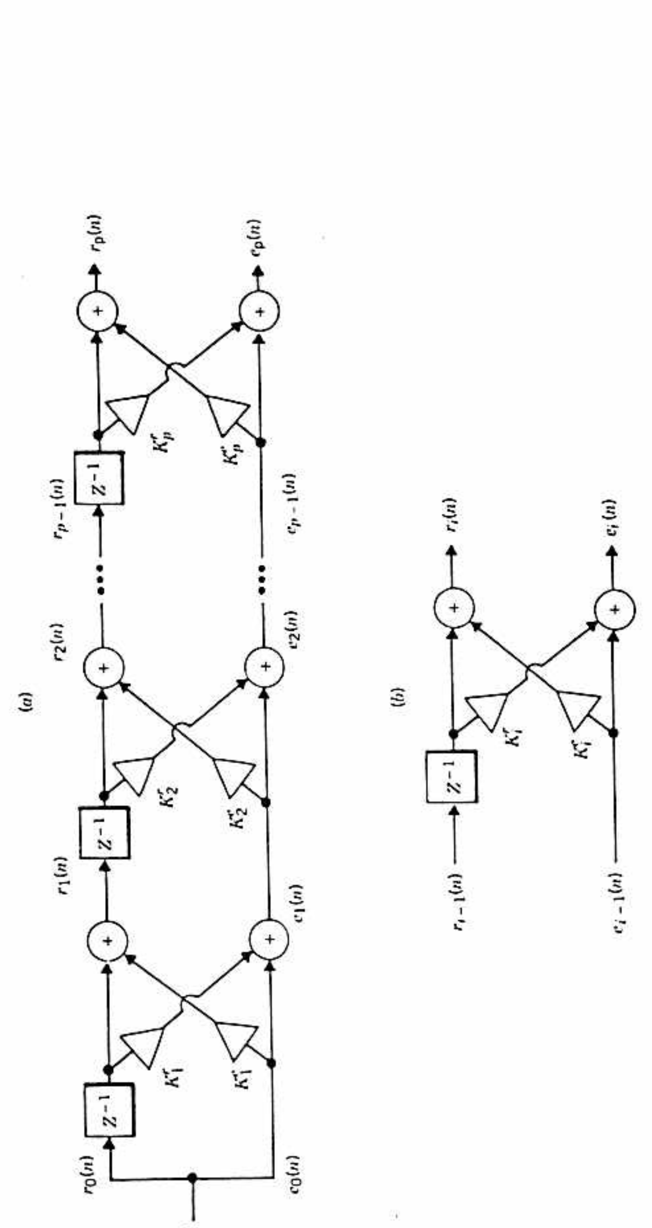


**Figure 11.4** (*a*) Forward and backward prediction error filters. (*b*) The *i*th stage of the lattice.





and

$$E^r_{p+1} = E^r_p\left(1 - K^e_{p+1}K^r_{p+1}\right) \tag{11.47}$$

Since $\Phi$ is Toeplitz,

$$E^r_{p+1} = E^e_{p+1} \tag{11.48}$$

$$= E^e_p\left(1 - |K^r_{p+1}|^2\right) \tag{11.49}$$

The order update sequence is described graphically in Figure 11.3. The forward and backward prediction error filters of order $p$ are updated using the weighting factors $K^r_{p+1}$ and $K^e_{p+1}$ to yield the corresponding prediction error filters of order $p + 1$. The lattice structure shown in Figure 11.4 is the result of building the forward and backward prediction error filters of order $p$ completely by this order update procedure.

### 11.2.2 Relationship between the Direct-Form and Lattice Prediction Error Filter Coefficients

As shown in Section 11.2.1, an order-recursive procedure evolves out of the solution to the one-step linear prediction problem when $\Phi$ is Toeplitz. This procedure is known as the Levinson–Durbin algorithm.[17,18] Summarizing (see Figs. 11.2 through 11.4 with $K_i = K^r_i = K^{e*}_i$ and $E_i = E^r_i = E^e_i$):

Initialization

$$E_0 = \phi_0 \tag{11.50a}$$

Order update ($i = 1, 2, \ldots, p$)

$$K_i = -\frac{1}{E_{i-1}}\sum_{k=0}^{i-1} a_k^{(i-1)}\phi_{i-k}, \quad a_0 = 1 \tag{11.50b}$$

$$a_i^{(i)} = K_i \tag{11.50c}$$

$$a_k^{(i)} = a_k^{(i-1)} + K_i a_{i-k}^{(i-1)*}, \quad 1 \le k \le i-1 \tag{11.50d}$$

$$E_i = \left(1 - |K_i|^2\right)E_{i-1} \tag{11.50e}$$

In the order-update recursion, Eqs. 11.50*c* and 11.50*d* are sometimes referred to as the step-up algorithm, where the equivalent direct form prediction error filter of order $p$ can be generated from a set of $p$ lattice coefficients.

The transfer functions of the forward and backward prediction error filters are defined as

$$A_p(z) = \sum_{k=0}^{p} a_k z^{-k}, \quad a_0 = 1 \tag{11.51a}$$

$$B_p(z) = \sum_{k=0}^{p} b_k z^{-k}, \quad b_p = 1 \tag{11.51b}$$

Since the forward and backward prediction error filters are related

$$\begin{bmatrix} b_0^p \\ \vdots \\ b_{p-1}^p \\ 1 \end{bmatrix} = \begin{bmatrix} a_p^{p*} \\ \vdots \\ a_1^{p*} \\ 1 \end{bmatrix} \tag{11.52}$$

their transfer functions also are related

$$B_p(z) = z^{-p} A_p^*\left(\frac{1}{z^*}\right) \tag{11.53}$$

Now, in terms of transfer functions, the step-up portion of the Levinson–Durbin algorithm can be expressed as

Initialization

$$A_0(z) = B_0(z) \tag{11.54a}$$

Order update ($i = 1, 2, \ldots, p$)

$$A_i(z) = A_{i-1}(z) + K_i z^{-1} B_{i-1}(z) \tag{11.54b}$$

$$B_i(z) = K_i^* A_{i-1}(z) + z^{-1} B_{i-1}(z) \tag{11.54c}$$

or, writing Eqs. 11.54$b$ and 11.54$c$ in matrix form

$$\begin{bmatrix} A_i(z) \\ B_i(z) \end{bmatrix} = \begin{bmatrix} 1 & z^{-1}K_i \\ K_i^* & z^{-1} \end{bmatrix} \begin{bmatrix} A_{i-1}(z) \\ B_{i-1}(z) \end{bmatrix} \tag{11.54d}$$

From Eq. 11.54$d$, the inverse procedure yields

$$A_{i-1}(z) = \frac{A_i(z) - K_i B_i(z)}{1 - |K_i|^2} \tag{11.55}$$

and the counterpart to Eqs. 11.50$c$ and 11.50$d$ is

Order update ($i = p, \ldots, 2, 1$)

$$K_i = a_i^{(i)} \tag{11.56a}$$

$$a_k^{(i-1)} = \frac{a_k^{(i)} - K_i a_{i-k}^{(i)*}}{1 - |K_i|^2}, \qquad 1 \le k \le i - 1 \tag{11.56b}$$

The recursion of Eq. 11.56 sometimes is referred to as the step-down algorithm where the equivalent lattice prediction error filter of order $p$ can be generated from a set of $p$ direct-form filter coefficients.



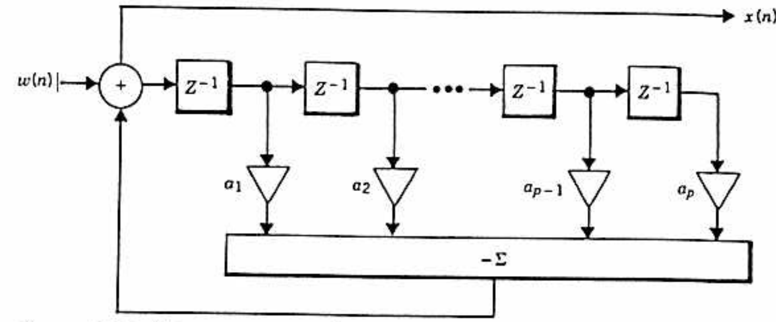


**Figure 11.5** AR process generation model.

### 11.2.3 Spectral Estimation

In recent years, a great deal of interest has been shown in high-resolution spectral estimation techniques.[7,9,11,13] This interest typically has been motivated by the desire to resolve narrowband frequency components in data records too short for adequate frequency separation via standard FFT techniques. By incorporating into the estimation problem assumptions about how the observed data were generated, rather remarkable results have been obtained.

For a number of physical reasons, modeling the observed data as an AR (autoregressive) process has been accepted widely in several applications of time series analysis. As shown in Figure 11.5, an AR process of order $p$ is obtained by passing the white noise sequence $x(n)$ having 0 mean and variance $\sigma_x^2$ through a $p$-pole filter. The corresponding input-output relationships are

$$\begin{aligned} x(n) &= -a_1 x(n-1) - a_2 x(n-2) - \cdots - a_p x(n-p) + w(n) \\ &= -\sum_{k=1}^{p} a_k x(n-k) + w(n) \end{aligned} \tag{11.57}$$

where $w(n)$ is called the innovation of the process. By evaluating the all-pole filter's $z$-transform on the unit circle, the power spectrum of the AR process $x(n)$ is obtained as

$$\begin{aligned} S_x(\omega) &= \left.\frac{\sigma_w^2}{|A_p(z)|^2}\right|_{z=e^{j\omega}} \\ &= \frac{\sigma_w^2}{\left|1 + \sum_{k=1}^{p} a_k \exp(-j\omega k)\right|^2} \end{aligned} \tag{11.58}$$

Notice that the all-zero filter of order $p$ that will recover $w(n)$ from $x(n)$ is $A_p(z)$. Appropriately, this filter has been called the whitening or inverse filter for the AR process and is simply obtained by the one-step forward linear predictor in Figure 11.2$a$. Removing the predictable components from $x(n)$ [or, correspondingly, the coloring from $S_x(\omega)$] yields the white forward prediction error sequence $e_p(n) = w(n)$. Now, given that the time series of interest is $p$th-order AR, the spectral estimation task becomes equivalently a problem of estimating the $p$th-order linear predictor, $A_p(z)$, along with the power, $\sigma_w^2$, of its prediction error output sequence.

### 11.2.4 General Filter Solution

The solution for the unknown filter vector $\mathbf{h}$ in Eq. 11.18 can be rewritten and augmented with the expression for minimum average squared error in Eq. 11.22 as follows.

$$\begin{bmatrix} \phi_0^d & g_0^* & \cdots & g_p^* \\ g_0 & \phi_0 & \cdots & \phi_p^* \\ \vdots & \vdots & & \vdots \\ g_0 & \phi_p & \cdots & \phi_0 \end{bmatrix} \begin{bmatrix} 1 \\ h_0^p \\ \vdots \\ h_p^p \end{bmatrix} = \begin{bmatrix} E_p^d \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.59}$$

Note the similarity of Eq. 11.59 and the corresponding expression in Eq. 11.30 for the forward prediction error filter.

Given the general filter of order $p$ has been determined, it now is of interest to derive an order update recursion that will permit generating the general filter of order $p+1$ from that of order $p$. Simply guess that the correct filter of order $p+1$ is $[1\ h_0^p \ldots h_p^p\ 0]^T$. Now, Eq. 11.59 becomes

$$\begin{bmatrix} \phi_0^d & g_0^* & \cdots & g_p^* & g_{p+1}^* \\ g_0 & \phi_0 & & \phi_p & \phi_{p+1} \\ \vdots & & & \vdots & \vdots \\ g_p & \phi_p & \cdots & \phi_0 & \phi_1^* \\ g_{p+1} & \phi_{p+1} & \cdots & \phi_1 & \phi_0 \end{bmatrix} \begin{bmatrix} 1 \\ h_0^p \\ \vdots \\ h_p^p \\ 0 \end{bmatrix} = \begin{bmatrix} E_p^d \\ 0 \\ \vdots \\ 0 \\ \Delta_{p+1}^d \end{bmatrix} \tag{11.60}$$

where

$$\Delta_{p+1}^d = g_{p+1} + \phi_{p+1} h_0^p + \cdots + \phi_1 h_p^p \tag{11.61}$$

If the discrepancy term $\Delta_{p+1}^d$ was 0, the proposed general filter of order $p+1$ would be correct. Since the discrepancy term generally is not 0, the next step involves modifying the proposed general filter coefficients so as to cancel the discrepancy. The modification involves adding a weighted version of the backward prediction error filter vector of order $p+1$ to the proposed general



filter vector

$$\begin{bmatrix} 1 \\ h_0^{p+1} \\ \vdots \\ h_p^{p+1} \\ h_{p+1}^{p+1} \end{bmatrix} = \begin{bmatrix} 1 \\ h_0^p \\ \vdots \\ h_p^p \\ 0 \end{bmatrix} + K_{p+1}^d \begin{bmatrix} 0 \\ b_0^{p+1} \\ \vdots \\ b_p^{p+1} \\ 1 \end{bmatrix} \tag{11.62}$$

yielding

$$\begin{bmatrix} E_{p+1}^d \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} E_p^d \\ 0 \\ \vdots \\ 0 \\ \Delta_{p+1}^d \end{bmatrix} + K_{p+1}^d \begin{bmatrix} \nabla_{p+1}^d \\ 0 \\ \vdots \\ 0 \\ E_{p+1}^r \end{bmatrix}. \tag{11.63}$$

where

$$\nabla_{p+1}^d = g_0^* b_0^{p+1} + \cdots + g_p^* b_p^{p+1} + g_{p+1}^* \tag{11.64}$$

In order to cancel the discrepancy term

$$K_{p+1}^d = -\frac{\Delta_{p+1}^d}{E_{p+1}^r} \tag{11.65}$$

Thus, the weighting factor consists of the discrepancy normalized by the backward prediction error from the prediction error filter of order $p+1$. Now, the update equation for average squared error is

$$E_{p+1}^d = E_p^d + K_{p+1}^d \nabla_{p+1}^d \tag{11.66}$$

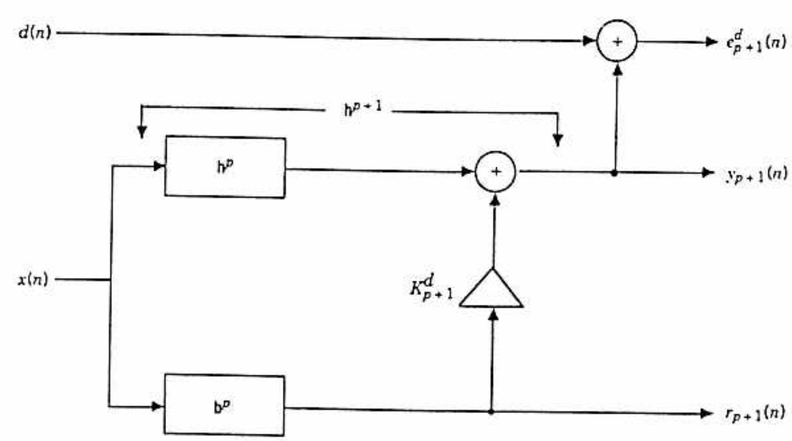


**Figure 11.6** Order update procedure for the general filter.

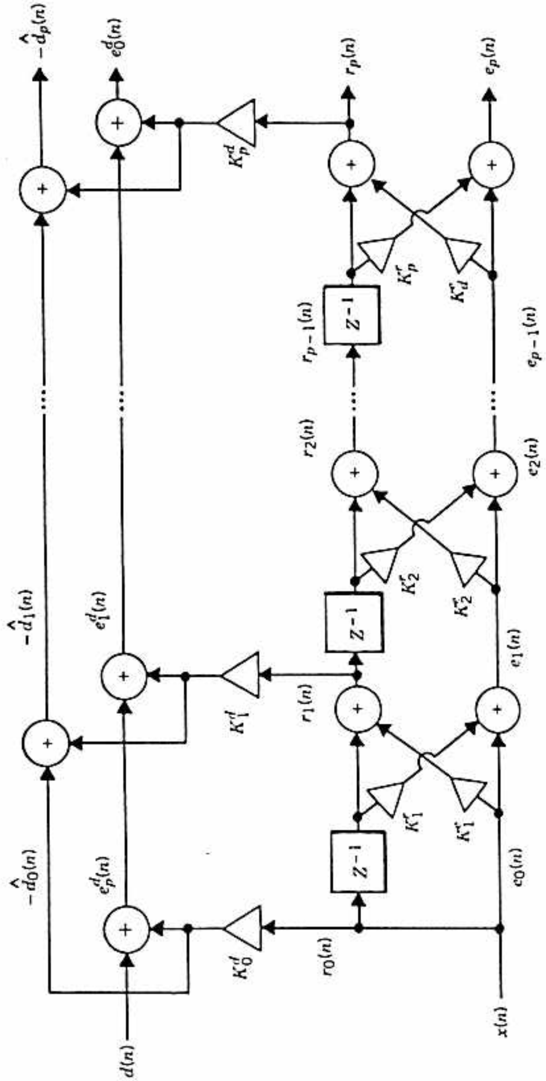


**Figure 11.7** Lattice realization of the joint process structure.





Since $\Phi$ is Toeplitz

$$\nabla_{p+1}^{d} = \Delta_{p+1}^{d*} \tag{11.67}$$

and by substituting Eq. 11.65 into the update equation for average squared error,

$$E_{p+1}^{d} = E_p^d + K_{p+1}^{d}\Delta_{p+1}^{d*} \tag{11.68}$$

$$= E_p^d - \left|K_{p+1}^{d}\right|^2 E_{p+1}^{r} \tag{11.69}$$

The order update sequence is described graphically in Figure 11.6. The general filter of order $p$ is updated using the weighting coefficient $K_{p+1}^{d}$ to yield the corresponding general filter of order $p+1$. The lattice structure shown in Figure 11.7 is the result of building the prediction error filters and the general filter of order $p$ completely by this order update procedure.

D. DeFatta, J. Lucas, and W. Hodgkiss.
Digital Signal Processing.
Wiley (1988).