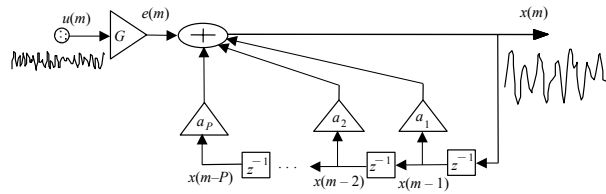


# 8

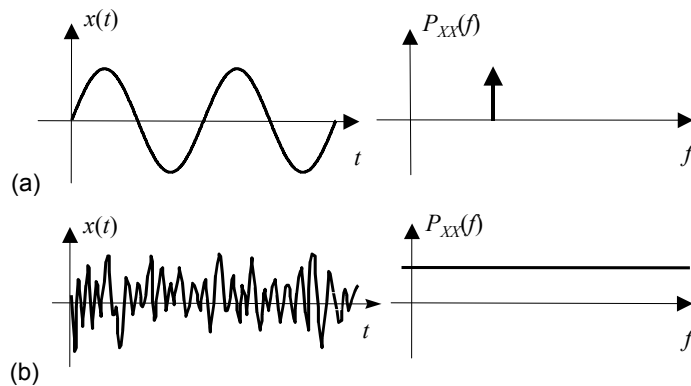


## LINEAR PREDICTION MODELS

- 8.1 Linear Prediction Coding
- 8.2 Forward, Backward and Lattice Predictors
- 8.3 Short-term and Long-Term Linear Predictors
- 8.4 MAP Estimation of Predictor Coefficients
- 8.5 Sub-Band Linear Prediction
- 8.6 Signal Restoration Using Linear Prediction Models
- 8.7 Summary

- 8.8 Applications
  - Speech/music Coding
  - Video Coding
  - Pattern Recognition
  - Radar Signal Processing
  - Watermarking
  - Noise Reduction
  - Interpolation/Smoothing
  - Change Detection

Linear prediction model is used in a diverse area of applications, such as data forecasting, speech coding, video coding, speech recognition, model-based spectral analysis, model-based signal interpolation, signal restoration, noise reduction, impulse detection, and change detection. In the statistical literature, linear prediction models are often referred to as autoregressive (AR) processes. In this chapter, we introduce the theory of linear prediction modelling and consider efficient methods for the computation of predictor coefficients. We study the forward, backward and lattice predictors, and consider various methods for the formulation and calculation of predictor coefficients, including the least square error and maximum a posteriori methods. For modeling quasi-periodic signals, such as voiced speech, an extended linear predictor that simultaneously utilizes the short and long-term correlation structures is introduced. We study sub-band linear predictors that are particularly useful for sub-band coding and processing of noisy signals. Finally, the application of linear prediction models in signal coding, enhancement, pattern recognition and watermarking are considered.



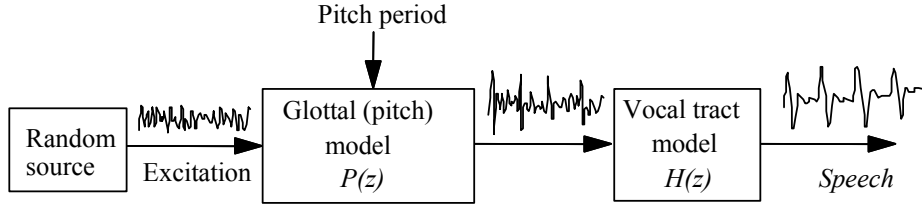
**Figure 8.1** The concentration or spread of power in frequency indicates the predictable or random character of a signal: (a) a predictable signal; (b) a random signal.

## 8.1 Linear Prediction Coding

The success with which a signal can be predicted from its past samples depends on the autocorrelation function, or equivalently the bandwidth and the power spectrum, of the signal. As illustrated in Figure 8.1, in the time domain a predictable signal has a smooth and correlated fluctuation, and in the frequency domain, the energy of a predictable signal is concentrated in narrow band/s of frequencies. In contrast, the energy of an unpredictable signal, such as a white noise, is spread over a wide band of frequencies.

For a signal to have a capacity to convey information it must have a degree of randomness. Most signals, such as speech, music and video signals, are partially predictable and partially random. These signals can be modelled as the output of a filter excited by an uncorrelated input. The random input models the unpredictable part of the signal, whereas the filter models the predictable structure of the signal. The aim of linear prediction is to model the mechanism that introduces the correlation in a signal.

Linear prediction models are extensively used in speech processing applications such as in low bit-rate speech coders, speech enhancement and speech recognition. Speech is generated by inhaling air and then exhaling it through the glottis and the vocal tract. The noise-like air, from the lung, is modulated and shaped by the vibrations of the glottal cords and the resonance of the vocal tract. Figure 8.2 illustrates a source-filter model of speech. The source models the lung, and emits a random input excitation signal that is filtered by a pitch filter. The pitch filter models the vibrations



**Figure 8.2** A source–filter model of speech production.

of the glottal cords, and generates a sequence of quasi-periodic excitation pulses for voiced sounds as shown in Figure 8.2. The pitch filter model is also termed the “long-term predictor” since it models the correlation of each sample with the samples a pitch period away. The main source of correlation and power amplification in speech is the vocal tract. The vocal tract is modelled by a linear predictor model, which is also termed the “short-term predictor”, because it models the correlation of each sample with the few (typically 8 to 20) preceding samples. In this section, we study the short-term linear prediction model. In Section 8.3, the predictor model is extended to include long-term pitch period correlations.

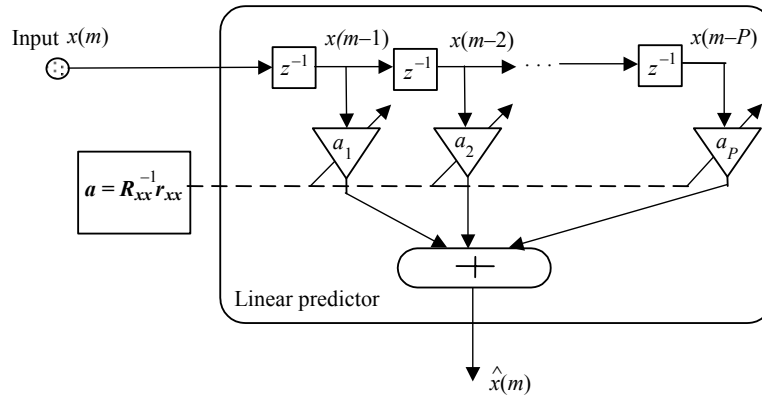
A linear predictor model forecasts the amplitude of a signal at time  $m$ ,  $x(m)$ , using a linearly weighted combination of  $P$  past samples [ $x(m-1)$ ,  $x(m-2)$ , ...,  $x(m-P)$ ] as

$$\hat{x}(m) = \sum_{k=1}^P a_k x(m-k) \quad (8.1)$$

where the integer variable  $m$  is the discrete time index,  $\hat{x}(m)$  is the prediction of  $x(m)$ , and  $a_k$  are the predictor coefficients. A block-diagram implementation of the predictor Equation (8.1) is illustrated in Figure 8.3.

The prediction error  $e(m)$ , defined as the difference between the actual sample value  $x(m)$  and its predicted value  $\hat{x}(m)$ , is given by

$$\begin{aligned} e(m) &= x(m) - \hat{x}(m) \\ &= x(m) - \sum_{k=1}^P a_k x(m-k) \end{aligned} \quad (8.2)$$



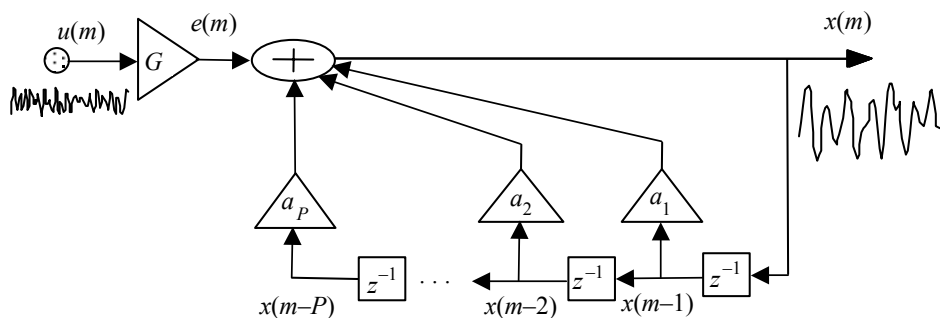
**Figure 8.3** Block-diagram illustration of a linear predictor.

For information-bearing signals, the prediction error  $e(m)$  may be regarded as the information, or the innovation (i.e. ‘new’), content of the sample  $x(m)$ . From Equation (8.2) a signal generated, or modelled, by a linear predictor can be described by the following feedback equation

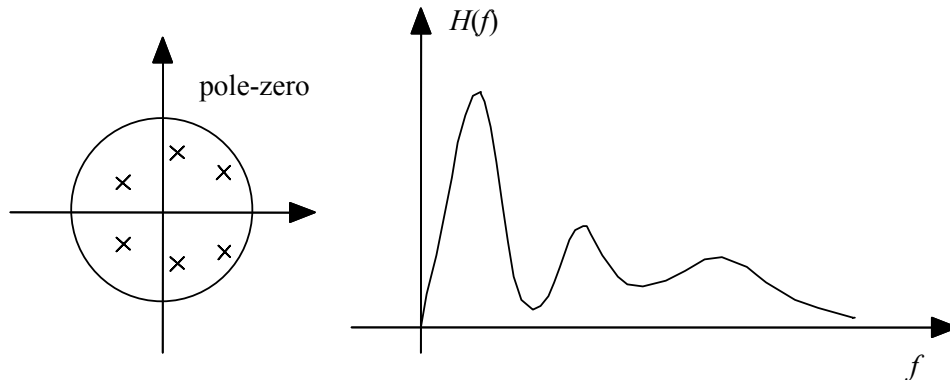
$$x(m) = \sum_{k=1}^P a_k x(m-k) + e(m) \quad (8.3)$$

Figure 8.4 illustrates a linear predictor model of a signal  $x(m)$ . In this model, the random input excitation (i.e. the prediction error) is  $e(m) = Gu(m)$ , where  $u(m)$  is a zero-mean, unit-variance random signal, and  $G$ , a gain term, is the square root of the variance (i.e. power) of  $e(m)$ :

$$G = (\mathbb{E}[e^2(m)])^{1/2} \quad (8.4)$$



**Figure 8.4** Illustration of a signal generated by a linear predictive model.



**Figure 8.5** The pole–zero position and frequency response of a linear predictor.

where  $\mathcal{E}[\cdot]$  is an averaging, or expectation, operator. Taking the  $z$ -transform of Equation (8.3) shows that the linear prediction model is an all-pole digital filter with  $z$ -transfer function

$$H(z) = \frac{X(z)}{U(z)} = \frac{G}{1 - \sum_{k=1}^P a_k z^{-k}} \quad (8.5)$$

In general, a linear predictor of order  $P$  has  $P/2$  complex pole pairs, and can model up to  $P/2$  resonance of the signal spectrum as illustrated in Figure 8.5. Spectral analysis using linear prediction models is discussed in Chapter 9.

### 8.1.1 Least Mean Square Error Predictor

The “best” predictor coefficients are obtained by minimising a mean square error criterion defined as

$$\begin{aligned}
\mathcal{E}[e^2(m)] &= \mathcal{E}\left[\left(x(m) - \sum_{k=1}^P a_k x(m-k)\right)^2\right] \\
&= \mathcal{E}[x^2(m)] - 2 \sum_{k=1}^P a_k \mathcal{E}[x(m)x(m-k)] + \sum_{k=1}^P a_k \sum_{j=1}^P a_j \mathcal{E}[x(m-k)x(m-j)] \\
&= r_{xx}(0) - 2\mathbf{r}_{xx}^T \mathbf{a} + \mathbf{a}^T \mathbf{R}_{xx} \mathbf{a}
\end{aligned} \tag{8.6}$$

where  $\mathbf{R}_{xx} = \mathcal{E}[\mathbf{x}\mathbf{x}^T]$  is the autocorrelation matrix of the input vector  $\mathbf{x}^T = [x(m-1), x(m-2), \dots, x(m-P)]$ ,  $\mathbf{r}_{xx} = \mathcal{E}[x(m)\mathbf{x}]$  is the autocorrelation vector and  $\mathbf{a}^T = [a_1, a_2, \dots, a_P]$  is the predictor coefficient vector. From Equation (8.6), the gradient of the mean square prediction error with respect to the predictor coefficient vector  $\mathbf{a}$  is given by

$$\frac{\partial}{\partial \mathbf{a}} \mathcal{E}[e^2(m)] = -2\mathbf{r}_{xx}^T + 2\mathbf{a}^T \mathbf{R}_{xx} \tag{8.7}$$

where the gradient vector is defined as

$$\frac{\partial}{\partial \mathbf{a}} = \left( \frac{\partial}{\partial a_1}, \frac{\partial}{\partial a_2}, \dots, \frac{\partial}{\partial a_P} \right)^T \tag{8.8}$$

The least mean square error solution, obtained by setting Equation (8.7) to zero, is given by

$$\mathbf{R}_{xx} \mathbf{a} = \mathbf{r}_{xx} \tag{8.9}$$

From Equation (8.9) the predictor coefficient vector is given by

$$\mathbf{a} = \mathbf{R}_{xx}^{-1} \mathbf{r}_{xx} \tag{8.10}$$

Equation (8.10) may also be written in an expanded form as

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_P \end{pmatrix} = \begin{pmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(P-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(P-2) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(P-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(P-1) & r_{xx}(P-2) & r_{xx}(P-3) & \cdots & r_{xx}(0) \end{pmatrix}^{-1} \begin{pmatrix} r_{xx}(1) \\ r_{xx}(2) \\ r_{xx}(3) \\ \vdots \\ r_{xx}(P) \end{pmatrix} \quad (8.11)$$

An alternative formulation of the least square error problem is as follows. For a signal block of  $N$  samples  $[x(0), \dots, x(N-1)]$ , we can write a set of  $N$  linear prediction error equations as

$$\begin{pmatrix} e(0) \\ e(1) \\ e(2) \\ \vdots \\ e(N-1) \end{pmatrix} = \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{pmatrix} - \begin{pmatrix} x(-1) & x(-2) & x(-3) & \cdots & x(-P) \\ x(0) & x(-1) & x(-2) & \cdots & x(1-P) \\ x(1) & x(0) & x(-1) & \cdots & x(2-P) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x(N-2) & x(N-3) & x(N-4) & \cdots & x(N-P-1) \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_P \end{pmatrix} \quad (8.12)$$

where  $\mathbf{x}^T = [x(-1), \dots, x(-P)]$  is the initial vector. In a compact vector/matrix notation Equation (8.12) can be written as

$$\mathbf{e} = \mathbf{x} - \mathbf{X}\mathbf{a} \quad (8.13)$$

Using Equation (8.13), the sum of squared prediction errors over a block of  $N$  samples can be expressed as

$$\mathbf{e}^T \mathbf{e} = \mathbf{x}^T \mathbf{x} - 2\mathbf{x}^T \mathbf{X}\mathbf{a} + \mathbf{a}^T \mathbf{X}^T \mathbf{X}\mathbf{a} \quad (8.14)$$

The least squared error predictor is obtained by setting the derivative of Equation (8.14) with respect to the parameter vector  $\mathbf{a}$  to zero:

$$\frac{\partial \mathbf{e}^T \mathbf{e}}{\partial \mathbf{a}} = -2\mathbf{x}^T \mathbf{X} + 2\mathbf{a}^T \mathbf{X}^T \mathbf{X} = 0 \quad (8.15)$$

From Equation (8.15), the least square error predictor is given by

$$\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{x}) \quad (8.16)$$

A comparison of Equations (8.11) and (8.16) shows that in Equation (8.16) the autocorrelation matrix and vector of Equation (8.11) are replaced by the time-averaged estimates as

$$\hat{r}_{xx}(m) = \frac{1}{N} \sum_{k=0}^{N-1} x(k)x(k-m) \quad (8.17)$$

Equations (8.11) and (8.16) may be solved efficiently by utilising the regular Toeplitz structure of the correlation matrix  $\mathbf{R}_{xx}$ . In a Toeplitz matrix, all the elements on a left–right diagonal are equal. The correlation matrix is also cross-diagonal symmetric. Note that altogether there are only  $P+1$  unique elements  $[r_{xx}(0), r_{xx}(1), \dots, r_{xx}(P)]$  in the correlation matrix and the cross-correlation vector. An efficient method for solution of Equation (8.10) is the Levinson–Durbin algorithm, introduced in Section 8.2.2.

### 8.1.2 The Inverse Filter: Spectral Whitening

The all-pole linear predictor model, in Figure 8.4, shapes the spectrum of the input signal by transforming an uncorrelated excitation signal  $u(m)$  to a correlated output signal  $x(m)$ . In the frequency domain the input–output relation of the all-pole filter of Figure 8.6 is given by

$$X(f) = \frac{G U(f)}{A(f)} = \frac{E(f)}{1 - \sum_{k=1}^P a_k e^{-j2\pi f k}} \quad (8.18)$$

where  $X(f)$ ,  $E(f)$  and  $U(f)$  are the spectra of  $x(m)$ ,  $e(m)$  and  $u(m)$  respectively,  $G$  is the input gain factor, and  $A(f)$  is the frequency response of the inverse predictor. As the excitation signal  $e(m)$  is assumed to have a flat spectrum, it follows that the shape of the signal spectrum  $X(f)$  is due to the frequency

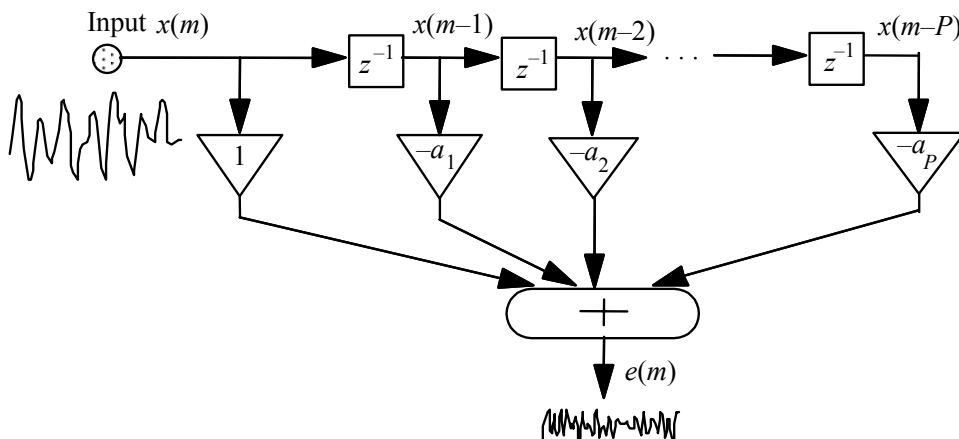


Figure 8.6 Illustration of the inverse (or whitening) filter.

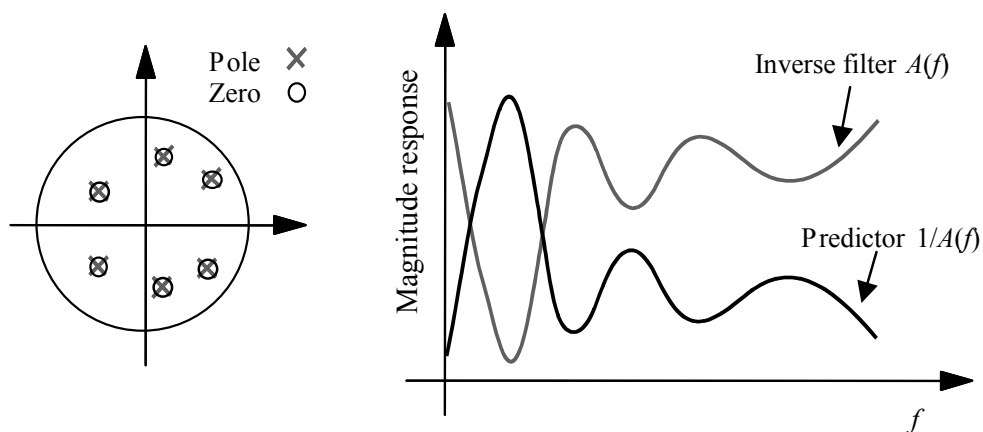
response  $1/A(f)$  of the all-pole predictor model. The inverse linear predictor, as the name implies, transforms a correlated signal  $x(m)$  back to an uncorrelated flat-spectrum signal  $e(m)$ . The inverse filter, also known as the prediction error filter, is an all-zero finite impulse response filter defined as

$$\begin{aligned} e(m) &= x(m) - \hat{x}(m) \\ &= x(m) - \sum_{k=1}^P a_k x(m-k) \\ &= (\mathbf{a}^{\text{inv}})^T \mathbf{x} \end{aligned} \quad (8.19)$$

where the inverse filter  $(\mathbf{a}^{\text{inv}})^T = [1, -a_1, \dots, -a_P] = [1, -\mathbf{a}]$ , and  $\mathbf{x}^T = [x(m), \dots, x(m-P)]$ . The  $z$ -transfer function of the inverse predictor model is given by

$$A(z) = 1 - \sum_{k=1}^P a_k z^{-k} \quad (8.20)$$

A linear predictor model is an all-pole filter, where the poles model the resonance of the signal spectrum. The inverse of an all-pole filter is an all-zero filter, with the zeros situated at the same positions in the pole-zero plot as the poles of the all-pole filter, as illustrated in Figure 8.7. Consequently, the zeros of the inverse filter introduce anti-resonances that cancel out the resonances of the poles of the predictor. The inverse filter has the effect of flattening the spectrum of the input signal, and is also known as a spectral whitening, or decorrelation, filter.



**Figure 8.7** Illustration of the pole-zero diagram, and the frequency responses of an all-pole predictor and its all-zero inverse filter.

### 8.1.3 The Prediction Error Signal

The prediction error signal is in general composed of three components:

- (a) the input signal, also called the excitation signal;
- (b) the errors due to the modelling inaccuracies;
- (c) the noise.

The mean square prediction error becomes zero only if the following three conditions are satisfied: (a) the signal is deterministic, (b) the signal is correctly modelled by a predictor of order  $P$ , and (c) the signal is noise-free. For example, a mixture of  $P/2$  sine waves can be modelled by a predictor of order  $P$ , with zero prediction error. However, in practice, the prediction error is nonzero because information bearing signals are random, often only approximately modelled by a linear system, and usually observed in noise. The least mean square prediction error, obtained from substitution of Equation (8.9) in Equation (8.6), is

$$E^{(P)} = \mathbb{E}[e^2(m)] = r_{xx}(0) - \sum_{k=1}^P a_k r_{xx}(k) \quad (8.21)$$

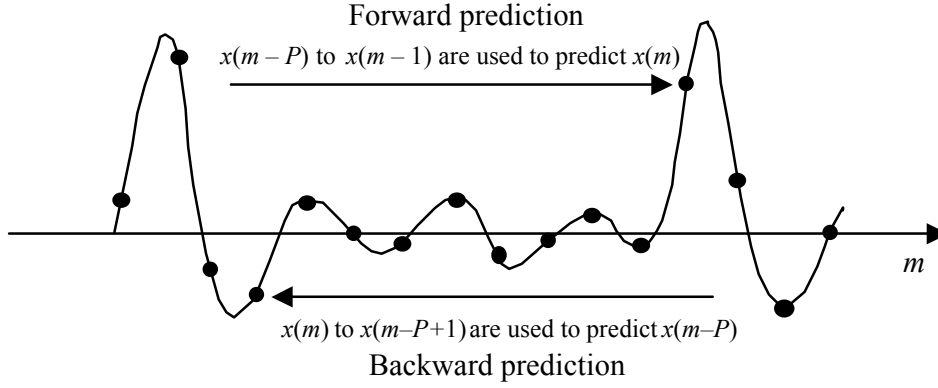
where  $E^{(P)}$  denotes the prediction error for a predictor of order  $P$ . The prediction error decreases, initially rapidly and then slowly, with increasing predictor order up to the correct model order. For the correct model order, the signal  $e(m)$  is an uncorrelated zero-mean random process with an autocorrelation function defined as

$$\mathbb{E}[e(m)e(m-k)] = \begin{cases} \sigma_e^2 = G^2 & \text{if } m = k \\ 0 & \text{if } m \neq k \end{cases} \quad (8.22)$$

where  $\sigma_e^2$  is the variance of  $e(m)$ .

## 8.2 Forward, Backward and Lattice Predictors

The forward predictor model of Equation (8.1) predicts a sample  $x(m)$  from a linear combination of  $P$  past samples  $x(m-1)$ ,  $x(m-2)$ , . . . ,  $x(m-P)$ .



**Figure 8.8** Illustration of forward and backward predictors.

Similarly, as shown in Figure 8.8, we can define a backward predictor, that predicts a sample  $x(m-P)$  from  $P$  future samples  $x(m-P+1), \dots, x(m)$  as

$$\hat{x}(m-P) = \sum_{k=1}^P c_k x(m-k+1) \quad (8.23)$$

The backward prediction error is defined as the difference between the actual sample and its predicted value:

$$\begin{aligned} b(m) &= x(m-P) - \hat{x}(m-P) \\ &= x(m-P) - \sum_{k=1}^P c_k x(m-k+1) \end{aligned} \quad (8.24)$$

From Equation (8.24), a signal generated by a backward predictor is given by

$$x(m-P) = \sum_{k=1}^P c_k x(m-k+1) + b(m) \quad (8.25)$$

The coefficients of the least square error backward predictor, obtained in a similar method to that of the forward predictor in Section 8.1.1, are given by

$$\begin{pmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(P-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(P-2) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(P-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(P-1) & r_{xx}(P-2) & r_{xx}(P-3) & \cdots & r_{xx}(0) \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_P \end{pmatrix} = \begin{pmatrix} r_{xx}(P) \\ r_{xx}(P-1) \\ r_{xx}(P-2) \\ \vdots \\ r_{xx}(1) \end{pmatrix} \quad (8.26)$$

Note that the main difference between Equations (8.26) and (8.11) is that the correlation vector on the right-hand side of the backward predictor, Equation (8.26) is upside-down compared with the forward predictor, Equation (8.11). Since the correlation matrix is Toeplitz and symmetric, Equation (8.11) for the forward predictor may be rearranged and rewritten in the following form:

$$\begin{pmatrix} r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \cdots & r_{xx}(P-1) \\ r_{xx}(1) & r_{xx}(0) & r_{xx}(1) & \cdots & r_{xx}(P-2) \\ r_{xx}(2) & r_{xx}(1) & r_{xx}(0) & \cdots & r_{xx}(P-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{xx}(P-1) & r_{xx}(P-2) & r_{xx}(P-3) & \cdots & r_{xx}(0) \end{pmatrix} \begin{pmatrix} a_P \\ a_{P-1} \\ a_{P-2} \\ \vdots \\ a_1 \end{pmatrix} = \begin{pmatrix} r_{xx}(P) \\ r_{xx}(P-1) \\ r_{xx}(P-2) \\ \vdots \\ r_{xx}(1) \end{pmatrix} \quad (8.27)$$

A comparison of Equations (8.27) and (8.26) shows that the coefficients of the backward predictor are the time-reversed versions of those of the forward predictor

$$\mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_P \end{pmatrix} = \begin{pmatrix} a_P \\ a_{P-1} \\ a_{P-2} \\ \vdots \\ a_1 \end{pmatrix} = \mathbf{a}^B \quad (8.28)$$

where the vector  $\mathbf{a}^B$  is the reversed version of the vector  $\mathbf{a}$ . The relation between the backward and forward predictors is employed in the Levinson–Durbin algorithm to derive an efficient method for calculation of the predictor coefficients as described in Section 8.2.2.

### 8.2.1 Augmented Equations for Forward and Backward Predictors

The inverse forward predictor coefficient vector is  $[1, -a_1, \dots, -a_P]=[1, -\mathbf{a}^T]$ . Equations (8.11) and (8.21) may be combined to yield a matrix equation for the inverse forward predictor coefficients:

$$\begin{pmatrix} r(0) & \mathbf{r}_{xx}^T \\ \mathbf{r}_{xx} & \mathbf{R}_{xx} \end{pmatrix} \begin{pmatrix} 1 \\ -\mathbf{a} \end{pmatrix} = \begin{pmatrix} E^{(P)} \\ \mathbf{0} \end{pmatrix} \quad (8.29)$$

Equation (8.29) is called the augmented forward predictor equation. Similarly, for the inverse backward predictor, we can define an augmented backward predictor equation as

$$\begin{pmatrix} \mathbf{R}_{xx} & \mathbf{r}_{xx}^B \\ \mathbf{r}_{xx}^{BT} & r(0) \end{pmatrix} \begin{pmatrix} -\mathbf{a}^B \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ E^{(P)} \end{pmatrix} \quad (8.30)$$

where  $\mathbf{r}_{xx}^T = [r_{xx}(1), \dots, r_{xx}(P)]$  and  $\mathbf{r}_{xx}^{BT} = [r_{xx}(P), \dots, r_{xx}(1)]$ . Note that the superscript BT denotes backward and transposed. The augmented forward and backward matrix Equations (8.29) and (8.30) are used to derive an order-update solution for the linear predictor coefficients as follows.

### 8.2.2 Levinson–Durbin Recursive Solution

The Levinson-Durbin algorithm was invented by N. Levinson in 1947 and modified by J. Durbin in 1959. The Levinson–Durbin algorithm is a recursive order-update method for calculation of linear predictor coefficients. A forward-prediction error filter of order  $i$  can be described in terms of the forward and backward prediction error filters of order  $i-1$  as

$$(8.31)$$

or in a more compact vector notation as

$$\begin{pmatrix} 1 \\ -\mathbf{a}^{(i)} \end{pmatrix} = \begin{pmatrix} 1 \\ -\mathbf{a}^{(i-1)} \\ 0 \end{pmatrix} + k_i \begin{pmatrix} 0 \\ -\mathbf{a}^{(i-1)B} \\ 1 \end{pmatrix} \quad (8.32)$$

where  $k_i$  is called the reflection coefficient. The proof of Equation (8.32) and the derivation of the value of the reflection coefficient for  $k_i$  follows shortly. Similarly, a backward prediction error filter of order  $i$  is described in terms of the forward and backward prediction error filters of order  $i-1$  as

$$\begin{pmatrix} -\mathbf{a}^{(i)B} \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -\mathbf{a}^{(i-1)B} \\ 1 \end{pmatrix} + k_i \begin{pmatrix} 1 \\ -\mathbf{a}^{(i-1)} \\ 0 \end{pmatrix} \quad (8.33)$$

To prove the order-update equation (8.32) (or alternatively Equation (8.33)), we multiply both sides of the equation by the  $(i+1) \times (i+1)$  augmented matrix  $\mathbf{R}_{xx}^{(i+1)}$  and use the equality

$$\mathbf{R}_{xx}^{(i+1)} = \begin{pmatrix} \mathbf{R}_{xx}^{(i)} & \mathbf{r}_{xx}^{(i)B} \\ \mathbf{r}_{xx}^{(i)BT} & r_{xx}(0) \end{pmatrix} = \begin{pmatrix} r_{xx}(0) & \mathbf{r}_{xx}^{(i)T} \\ \mathbf{r}_{xx}^{(i)} & \mathbf{R}_{xx}^{(i)} \end{pmatrix} \quad (8.34)$$

to obtain

$$\begin{pmatrix} \mathbf{R}_{xx}^{(i)} & \mathbf{r}_{xx}^{(i)B} \\ \mathbf{r}_{xx}^{(i)BT} & r_{xx}(0) \end{pmatrix} \begin{pmatrix} 1 \\ -\mathbf{a}^{(i)} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_{xx}^{(i)} & \mathbf{r}_{xx}^{(i)B} \\ \mathbf{r}_{xx}^{(i)BT} & r_{xx}(0) \end{pmatrix} \begin{pmatrix} 1 \\ -\mathbf{a}^{(i-1)} \\ 0 \end{pmatrix} + k_i \begin{pmatrix} r_{xx}(0) & \mathbf{r}_{xx}^{(i)T} \\ \mathbf{r}_{xx}^{(i)} & \mathbf{R}_{xx}^{(i)} \end{pmatrix} \begin{pmatrix} 0 \\ -\mathbf{a}^{(i-1)B} \\ 1 \end{pmatrix} \quad (8.35)$$

where in Equation (8.34) and Equation (8.35)  $\mathbf{r}_{xx}^{(i)T} = [r_{xx}(1), \dots, r_{xx}(i)]$ , and  $\mathbf{r}_{xx}^{(i)BT} = [r_{xx}(i), \dots, r_{xx}(1)]$  is the reversed version of  $\mathbf{r}_{xx}^{(i)T}$ . Matrix-vector multiplication of both sides of Equation (8.35) and the use of Equations (8.29) and (8.30) yields

$$\begin{pmatrix} E^{(i)} \\ \mathbf{0}^{(i)} \end{pmatrix} = \begin{pmatrix} E^{(i-1)} \\ \mathbf{0}^{(i-1)} \\ \Delta^{(i-1)} \end{pmatrix} + k_i \begin{pmatrix} \Delta^{(i-1)} \\ \mathbf{0}^{(i-1)} \\ E^{(i-1)} \end{pmatrix} \quad (8.36)$$

where

$$\begin{aligned} \Delta^{(i-1)} &= \begin{bmatrix} 1 & -\mathbf{a}^{(i-1)} \end{bmatrix}^T \mathbf{r}_{xx}^{(i)B} \\ &= r_{xx}(i) - \sum_{k=1}^{i-1} a_k^{(i-1)} r_{xx}(i-k) \end{aligned} \quad (8.37)$$

If Equation (8.36) is true, it follows that Equation (8.32) must also be true. The conditions for Equation (8.36) to be true are

$$E^{(i)} = E^{(i-1)} + k_i \Delta^{(i-1)} \quad (8.38)$$

and

$$0 = \Delta^{(i-1)} + k_i E^{(i-1)} \quad (8.39)$$

From (8.39),

$$k_i = -\frac{\Delta^{(i-1)}}{E^{(i-1)}} \quad (8.40)$$

Substitution of  $\Delta^{(i-1)}$  from Equation (8.40) into Equation (8.38) yields

$$\begin{aligned} E^{(i)} &= E^{(i-1)}(1 - k_i^2) \\ &= E^{(0)} \prod_{j=1}^i (1 - k_j^2) \end{aligned} \quad (8.41)$$

Note that it can be shown that  $\Delta^{(i)}$  is the cross-correlation of the forward and backward prediction errors:

$$\Delta^{(i-1)} = \mathcal{E}[b^{(i-1)}(m-1)e^{(i-1)}(m)] \quad (8.42)$$

The parameter  $\Delta^{(i-1)}$  is known as the partial correlation.

### Levinson-Durbin algorithm

Equations (8.43)–(8.48) are solved recursively for  $i=1, \dots, P$ . The Durbin algorithm starts with a predictor of order zero for which  $E^{(0)}=r_{xx}(0)$ . The algorithm then computes the coefficients of a predictor of order  $i$ , using the coefficients of a predictor of order  $i-1$ . In the process of solving for the coefficients of a predictor of order  $P$ , the solutions for the predictor coefficients of all orders less than  $P$  are also obtained:

$$E^{(0)}=r_{xx}(0) \quad (8.43)$$

For  $i=1, \dots, P$

$$\Delta^{(i-1)}=r_{xx}(i)-\sum_{k=1}^{i-1} a_k^{(i-1)} r_{xx}(i-k) \quad (8.44)$$

$$k_i=-\frac{\Delta^{(i-1)}}{E^{(i-1)}} \quad (8.45)$$

$$a_i^{(i)}=-k_i \quad (8.46)$$

$$a_j^{(i)}=a_j^{(i-1)}+k_i a_{i-j}^{(i-1)} \quad 1 \leq j \leq i-1 \quad (8.47)$$

$$E^{(i)}=(1-k_i^2)E^{(i-1)} \quad (8.48)$$