CHAPTER

11

Method of Least Squares

In this chapter, we use a model-dependent procedure known as the *method of least squares* to solve the linear filtering problem, without invoking assumptions on the statistics of the inputs applied to the filter. To illustrate the basic idea of least squares, suppose we have a set of real-valued measurements $u(1), u(2), \ldots, u(N)$, made at times $t_1, t_2, \ldots, t_N$, respectively, and the requirement is to construct a curve that is used to *fit* these points in some optimum fashion. Let the time dependence of this curve be denoted by $f(t_i)$. According to the method of least squares, the "best" fit is obtained by *minimizing the sum of squares of difference* between $f(t_i)$ and $u(i)$ for $i = 1, 2, \ldots, N$, hence the name of the method.

The method of least squares may be viewed as an alternative to Wiener filter theory. Basically, Wiener filters are derived from *ensemble averages* with the result that one filter (optimum in a probabilistic sense) is obtained for all realizations of the operational environment, assumed to be wide-sense stationary. On the other hand, the method of least squares is *deterministic* in approach. Specifically, it involves the use of time averages, with the result that the filter depends on the number of samples used in the computation. We begin our study in the next section by outlining the essence of the linear least-squares estimation problem.

11.1 STATEMENT OF THE LINEAR LEAST-SQUARES ESTIMATION PROBLEM

Consider a physical phenomenon that is characterized by two sets of variables, $d(i)$ and $u(i)$. The variable $d(i)$ is observed at time i in *response* to the subset of variables $u(i)$, $u(i-1), \ldots, u(i-M+1)$ applied as *inputs*. That is, $d(i)$ is a function of the inputs $u(i)$,

$u(i-1), \ldots, u(i-M+1)$. This functional relationship is hypothesized to be *linear*. In particular, the response $d(i)$ is modeled as

$$d(i) = \sum_{k=0}^{M-1} w_{ok}^* u(i-k) + e_o(i) \tag{11.1}$$

where the w_{ok} are *unknown parameters* of the *model*, and $e_o(i)$ represents the *measurement error* to which the statistical nature of the phenomenon is ascribed; each term in the summation in Eq. (11.1) represents a scalar inner product. In effect, the model of Eq. (11.1) says that the variable $d(i)$ may be determined as a linear combination of the input variables $u(i), u(i-1), \ldots, u(i-M+1)$, except for the error $e_o(i)$. This model, represented by the signal-flow graph shown in Fig. 11.1, is called a *multiple linear regression model*.

The *measurement error* $e_o(i)$ is an *unobservable* random variable that is introduced into the model to account for its inaccuracy. It is customary to assume that the measurement error process $e_o(i)$ is white with zero mean and variance σ^2. That is,

$$E[e_o(i)] = 0 \qquad \text{for all } i$$

and

$$E[e_o(i)e_o^*(k)] = \begin{cases} \sigma^2, & i = k \\ 0, & i \neq k \end{cases}$$

The implication of this assumption is that we may rewrite Eq. (11.1) in the ensemble-averaged form

$$E[d(i)] = \sum_{k=0}^{M-1} w_{ok}^* u(i-k)$$

where the values of $u(i), u(i-1), \ldots, u(i-M+1)$ are known. Hence, the mean of the response $d(i)$, in theory, is uniquely determined by the model.

The problem we have to solve is to *estimate* the unknown parameters of the multiple linear regression model of Fig. 11.1, the w_{ok}, given the two *observable* sets of variables: $u(i)$ and $d(i)$, $i = 1, 2, \ldots, N$. To do this, we postulate the linear transversal filter of Fig. 11.2 as the model of interest. By forming inner scalar products of the *tap inputs* $u(i), u(i-1), \ldots, u(i-M+1)$ and the corresponding *tap weights* $w_0, w_1, \ldots, w_{M-1}$, and by utilizing $d(i)$ as the *desired response*, we define the *estimation error* or *residual* $e(i)$ as the difference between the desired response $d(i)$ and the *filter output* $y(i)$, as shown by

$$e(i) = d(i) - y(i) \tag{11.2}$$

where

$$y(i) = \sum_{k=0}^{M-1} w_k^* u(i-k) \tag{11.3}$$

That is,

$$e(i) = d(i) - \sum_{k=0}^{M-1} w_k^* u(i-k) \tag{11.4}$$

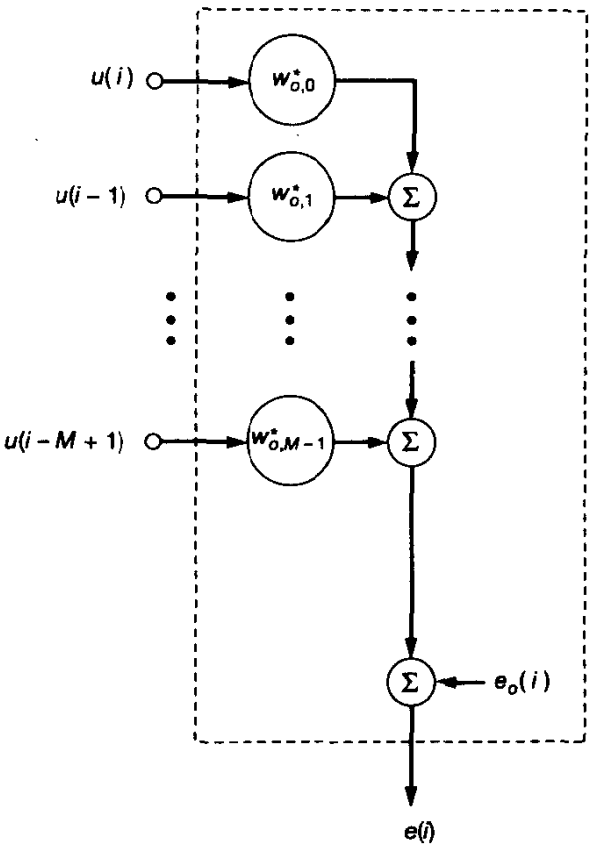

Figure 11.1 Multiple linear regression model.

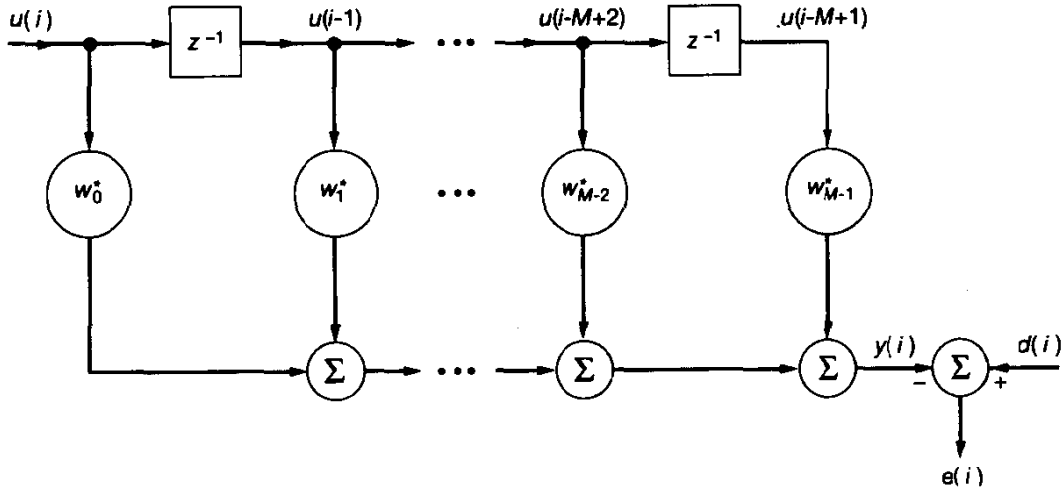

Figure 11.2 Linear transversal filter model.

In the method of least squares, we choose the tap weights of the transversal filter, the w_k, so as to minimize a cost function that consists of the *sum of error squares*:

$$\mathscr{E}(w_0, \ldots, w_{M-1}) = \sum_{i=i_1}^{i_2} |e(i)|^2 \tag{11.5}$$

where i_1 and i_2 define the index limits at which the error minimization occurs; this sum may also be viewed as an *error energy*. The values assigned to these limits depend on the type of *data windowing* employed, as discussed in Section 11.2. Basically, the problem we have to solve is to substitute Eq. (11.4) into (11.5) and then minimize the cost function $\mathscr{E}(w_0, \ldots, w_{M-1})$ with respect to the tap weights of the transversal filter in Fig. 11.2. For this minimization, the tap weights of the filter $w_0, w_1, \ldots, w_{M-1}$ are held *constant* during the interval $i_1 \leq i \leq i_2$. The filter resulting from the minimization is termed a *linear least-squares filter*.

11.2 DATA WINDOWING

Given M as the number of tap weights used in the transversal filter model of Fig. 11.2, the rectangular matrix constructed from the input data, $u(1), u(2), \ldots, u(N)$, may assume different forms, depending on the values assigned to the limits i_1 and i_2 in Eq. (11.5). In particular, we may distinguish four different methods of *windowing* the input data:

1. *Covariance method*, which makes no assumptions about the data outside the interval $[1, N]$. Thus, by defining the limits of interest as $i_1 = M$ and $i_2 = N$, the input data may be arranged in the matrix form

$$\begin{bmatrix} u(M) & u(M+1) & \cdots & u(N) \\ u(M-1) & u(M) & \cdots & u(N-1) \\ \vdots & \vdots & \vdots & \vdots \\ u(1) & u(2) & \cdots & u(N-M+1) \end{bmatrix}$$

2. *Autocorrelation method*, which makes the assumption that the data prior to time $i = 1$ and the data after $i = N$ are zero. Thus, by using $i_1 = 1$ and $i_2 = N + M - 1$, the matrix of input data takes on the form

$$\begin{bmatrix} u(1) & u(2) & \cdots & u(M) & u(M+1) & \cdots & u(N) & 0 & \cdots & 0 \\ 0 & u(1) & \cdots & u(M-1) & u(M) & \cdots & u(N-1) & u(N) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & u(1) & u(2) & \cdots & u(N-M+1) & u(N-M) & \cdots & u(N) \end{bmatrix}$$

3. *Prewindowing method*, which makes the assumption that the input data prior to $i = 1$ are zero, but makes no assumption about the data after $i = N$. Thus, by using $i_1 = 1$ and $i_2 = N$, the matrix of the input data assumes the form

$$\begin{bmatrix} u(1) & u(2) & \cdots & u(M) & u(M+1) & \cdots & u(N) \\ 0 & u(1) & \cdots & u(M-1) & u(M) & \cdots & u(N-1) \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & u(1) & u(2) & \cdots & u(N-M+1) \end{bmatrix}$$

4. *Postwindowing method*, which makes no assumption about the data prior to time $i = 1$, but makes the assumption that the data after $i = N$ are zero. Thus, by using $i_1 = M$ and $i_2 = N + M - 1$, the matrix of input data takes on the form

$$\begin{bmatrix} u(M) & u(M+1) & \cdots & u(N) & 0 & \cdots & 0 \\ u(M-1) & u(M) & \cdots & u(N-1) & u(N) & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ u(1) & u(2) & \cdots & u(N-M+1) & u(N-M) & \cdots & u(N) \end{bmatrix}$$

The terms "covariance method" and "autocorrelation method" are commonly used in speech-processing literature (Makhoul, 1975; Markel and Gray, 1976). It should, however, be emphasized that the use of these two terms is *not* based on the standard definition of the covariance function as the correlation function with the means removed. Rather, these two terms derive their names from the way we interpret the meaning of the *known parameters* contained in the system of equations that result from minimizing the index of performance of Eq. (11.5). The covariance method derives its name from control theory literature where, with zero-mean tap inputs, these known parameters represent the elements of a *covariance matrix*, hence the name of the method. The autocorrelation method, on the other hand, derives its name from the fact that, for the conditions stated, these known parameters represent the *short-term autocorrelation function* of the tap inputs, hence the name of the second method. It is of interest to note that, among the four windowing methods described above, the autocorrelation method is the only one that yields a *Toeplitz* correlation matrix for the input data.

In the remainder of this chapter, except for Problem 4, which deals with the autocorrelation method, we will be exclusively concerned with the covariance method. The prewindowing method is considered in subsequent chapters.

11.3 PRINCIPLE OF ORTHOGONALITY (REVISITED)

When we developed the Wiener filter theory in Chapter 5, we proceeded by first deriving the principle of orthogonality (in the ensemble sense) for wide-sense stationary discrete-

time stochastic processes, which were then used to derive the Wiener–Hopf equations that provide the mathematical basis of Wiener filters. In this chapter we proceed in a similar fashion by first deriving the principle of orthogonality based on time averages, and then use it to derive a system of equations known as the normal equations that provides the mathematical basis of linear least-squares filters. The development of this theory will be done for the covariance method.

The cost function or the sum of the error squares in the covariance method is defined by

$$\mathscr{E}(w_0, \ldots, w_{M-1}) = \sum_{i=M}^{N} |e(i)|^2 \tag{11.6}$$

By choosing the limits on the time index i in this way, in effect, we make sure that for each value of i, all the M tap inputs of the transversal filter in Fig. 11.2 have nonzero values. As mentioned previously, the problem we have to solve is to determine the tap weights of the transversal filter of Fig. 11.2 for which the sum of error squares is minimum.

We first rewrite Eq. (11.6) as

$$\mathscr{E}(w_0, \ldots, w_{M-1}) = \sum_{i=M}^{N} e(i)e^*(i) \tag{11.7}$$

where the estimation error $e(i)$ is defined in Eq. (11.4). Let the kth tap-weight w_k be expressed in terms of its real and imaginary parts as follows:

$$w_k = a_k + jb_k, \qquad k = 0, 1, \ldots, M-1 \tag{11.8}$$

Thus, substituting Eq. (11.8) in (11.4), we get

$$e(i) = d(i) - \sum_{k=0}^{M-1} (a_k - jb_k)u(i-k) \tag{11.9}$$

We define the kth component of the gradient vector $\nabla\mathscr{E}$ as the derivative of the cost function $\mathscr{E}(w_0, \ldots, w_{M-1})$ with respect to the real and imaginary parts of tap-weight w_k, as shown by

$$\nabla_k\mathscr{E} = \frac{\partial\mathscr{E}}{\partial a_k} + j\frac{\partial\mathscr{E}}{\partial b_k} \tag{11.10}$$

Hence, substituting Eq. (11.7) in (11.10), and recognizing that the estimation error $e(i)$ is complex valued, in general, we get

$$\nabla_k\mathscr{E} = -\sum_{i=M}^{N}\left[e(i)\frac{\partial e^*(i)}{\partial a_k} + e^*(i)\frac{\partial e(i)}{\partial a_k} + je(i)\frac{\partial e^*(i)}{\partial b_k} + je(i)\frac{\partial e(i)}{\partial b_k}\right] \tag{11.11}$$

Next, differentiating $e(i)$ in Eq. (11.9) with respect to the real and imaginary parts of w_k, we get the following four partial derivatives:

$$\begin{aligned}
\frac{\partial e(i)}{\partial a_k} &= -u(i-k) \\
\frac{\partial e^*(i)}{\partial a_k} &= -u^*(i-k) \\
\frac{\partial e(i)}{\partial b_k} &= ju(i-k) \\
\frac{\partial e^*(i)}{\partial b_k} &= -ju^*(i-k)
\end{aligned} \tag{11.12}$$

Thus, the substitution of these four partial derivatives in Eq. (11.11) yields the result:

$$\nabla_k \mathscr{E} = -2 \sum_{i=M}^{N} u(i-k)e^*(i) \tag{11.13}$$

For the minimization of the cost function $\mathscr{E}(w_0, \ldots, w_{M-1})$ with respect to the tap weights $w_0, \ldots, w_{M-1}$ of the transversal filter in Fig. 11.2, we require that the following conditions be satisfied simultaneously:

$$\nabla_k \mathscr{E} = 0, \qquad k = 0, 1, \ldots, M-1 \tag{11.14}$$

Let $e_{\min}(i)$ denote the special value of the estimation error $e(i)$ that results when the cost function $\mathscr{E}(w_0, \ldots, W_{M-1})$ is minimized (i.e., the transversal filter is optimized) in accordance with Eq. (11.14). From Eq. (11.13) we then readily see that the set of conditions (11.14) is equivalent to the following:

$$\sum_{i=M}^{N} u(i-k)e^*_{\min}(i) = 0, \qquad k = 0, 1, \ldots, M-1 \tag{11.15}$$

Equation (11.15) is the mathematical description of the temporal version of the *principle of orthogonality*. The *time average*[1] on the left-hand side of Eq. (11.15) represents the cross-correlation between the tap input $u(i-k)$ and the minimum estimation error $e_{\min}(i)$ over the values of time i in the interval $[M, N]$, for a fixed value of k. Accordingly, we may state the *principle of orthogonality* as follows:

The minimum error time series $e_{\min}(i)$ is orthogonal to the time series $u(i-k)$ applied to tap k of a transversal filter of length M for $k = 0, 1, \ldots, M-1$, when the filter is operating in its least-squares condition.

This principle provides the basis of a simple *test* that we can carry out in practice to check whether or not the transversal filter is operating in its *least-square condition*. We

[1]To be precise in the use of the term "time average," we should divide the sum on the left-hand side of Eq. (11.15) by the number of terms $(N - M + 1)$ used in the summation. Clearly, such an operation has no effect on Eq. (11.15). We have chosen to ignore the inclusion of this scaling factor merely for convenience of presentation.

merely have to determine the time-averaged cross-correlation between the estimation error and the time series applied to *each* tap input of the filter. It is *only* when *all* these M cross-correlation functions are identically zero that we find the cost function $\mathscr{E}(w_0, \ldots, w_{M-1})$ is minimum.

Corollary

Let $\hat{w}_0, \hat{w}_1, \ldots, \hat{w}_{M-1}$ denote the special values of the tap weights $w_0, w_1, \ldots, w_{M-1}$ that result when the transversal filter of Fig. 11.2 is optimized to operate in its least-squares condition. The filter output, denoted by $y_{\min}(i)$, is obtained from Eq. (11.3) to be

$$y_{\min}(i) = \sum_{k=0}^{M-1} \hat{w}_k^* u(i-k) \tag{11.16}$$

This filter output provides a *least-squares estimate* of the desired response $d(i)$; the estimate is said to be *linear* because it is a linear combination of the tap inputs $u(i), u(i-1), \ldots, u(i-M+1)$. Let $\mathscr{U}_i$ denote the space spanned by the tap inputs $u(i), \ldots, u(i-M+1)$. Let $\hat{d}(i|\mathscr{U}_i)$ denote the least-squares estimate of the desired response $d(i)$, given the tap inputs spanned by the space $\mathscr{U}_i$. We may thus write

$$\hat{d}(i|\mathscr{U}_i) = y_{\min}(i) \tag{11.17}$$

or, equivalently,

$$\hat{d}(i|\mathscr{U}_i) = \sum_{k=0}^{M-1} \hat{w}_k^* u(i-k) \tag{11.18}$$

Returning to Eq. (11.15), suppose we multiply both sides of this equation by $\hat{w}_k^*$ and then sum the result over the values of k in the interval $[0, M-1]$. We then get (after interchanging the order of summation):

$$\sum_{i=M}^{N} \left[\sum_{k=0}^{M-1} \hat{w}_k^* u(i-k) \right] e_{\min}^*(i) = 0 \tag{11.19}$$

The summation term inside the parentheses on the left-hand side of Eq. (11.19) is recognized to be the least-squares estimate $\hat{d}(i|\mathscr{U}_i)$ of Eq. (11.18). Accordingly, we may simplify Eq. (11.19) to

$$\sum_{i=M}^{N} \hat{d}(i|\mathscr{U}_i) e_{\min}^*(i) = 0 \tag{11.20}$$

Equation (11.20) is a mathematical description of the *corollary to the principle of orthogonality*. We recognize the time average on the left-hand side of Eq. (11.20) is the cross-correlation of the two time series $\hat{d}(i|\mathscr{U}_i)$ and $e_{\min}(i)$. Accordingly, we may state the corollary to the principle of orthogonality as follows:

When a transversal filter operates in its least-squares condition, the least-squares estimate of the desired response, produced at the filter output and represented by the time series

$\hat{d}(i|\mathcal{U}_i)$, and the minimum estimation error time series $e_{\min}(i)$ are orthogonal to each other over time i.

A geometric illustration of this corollary to the principle of orthogonality is deferred to Section 11.6.

11.4 MINIMUM SUM OF ERROR SQUARES

The principle of orthogonality, given in Eq. (11.15), describes the least-squares condition of the transversal filter in Fig. 11.2 when the cost function $\mathscr{E}(w_0, \ldots, w_{M-1})$ is minimized with respect to the tap weights $w_0, \ldots, w_{M-1}$ in the filter. To find the minimum value of this cost function, that is, the *minimum sum of error squares* $\mathscr{E}_{\min}$, it is obvious that we may write

$$\underbrace{d(i)}_{\substack{\text{desired}\\ \text{response}}} = \underbrace{\hat{d}(i|\mathcal{U}_i)}_{\substack{\text{estimate of}\\ \text{desired}\\ \text{response}}} + \underbrace{e_{\min}(i)}_{\substack{\text{estimation}\\ \text{error}}} \tag{11.21}$$

Hence, evaluating the energy of the time series $d(i)$ for values of time i in the interval $[M, N]$, and using the corollary to the principle of orthogonality [i.e., Eq. (11.20)], we get the simple result

$$\mathscr{E}_d = \mathscr{E}_{\text{est}} + \mathscr{E}_{\min} \tag{11.22}$$

where

$$\mathscr{E}_d = \sum_{i=M}^{N} |d(i)|^2 \tag{11.23}$$

$$\mathscr{E}_{\text{est}} = \sum_{i=M}^{N} |\hat{d}(i|\mathcal{U}_i)|^2 \tag{11.24}$$

$$\mathscr{E}_{\text{est}} = \sum_{i=M}^{N} |e_{\min}(i)|^2 \tag{11.25}$$

Rearranging Eq. (11.22), we may express the minimum sum of error squares $\mathscr{E}_{\min}$ in terms of the energy $\mathscr{E}_d$ and the energy $\mathscr{E}_{\text{est}}$, contained in the time series $d(i)$ and $\hat{d}(i|\mathcal{U}_i)$, respectively, as follows:

$$\mathscr{E}_{\min} = \mathscr{E}_d - \mathscr{E}_{\text{est}} \tag{11.26}$$

Clearly, given the specification of the desired response $d(i)$ for varying i, we may use Eq. (11.23) to evaluate the energy $\mathscr{E}_d$. As for the energy $\mathscr{E}_{\text{est}}$ contained in the time series $\hat{d}(i|\mathcal{U}_i)$ representing the estimate of the desired response, we are going to defer its evaluation to the next section.

Since $\mathscr{E}_{\min}$ is nonnegative, it follows that the second term on the right-hand side of Eq. (11.26) can never exceed $\mathscr{E}_d$. Indeed, it reaches the value of $\mathscr{E}_d$ when the measurement error $e_o(i)$ in the multiple linear regression model of Fig. 11.1 is zero for all i, which is a practical impossibility.

Another case for which $\mathscr{E}_{\min}$ equals $\mathscr{E}_d$ occurs when the least-squares problem is *underdetermined*. Such a situation arises when there are fewer data points than parameters, in which case the estimation error and therefore $\mathscr{E}_{\text{est}}$ is zero. Note, however, that when the least-squares problem is underdetermined, there is no unique solution to the problem. Discussion of this issue is deferred to the latter part of the chapter.

11.5 NORMAL EQUATIONS AND LINEAR LEAST-SQUARES FILTERS

There are two different, and yet basically equivalent, methods of describing the least-squares condition of the linear transversal filter in Fig. 11.1. The principle of orthogonality, described in Eq. (11.15), represents one method. The system of *normal equations* represents the other method; interestingly enough, the system of normal equations derives its name from the corollary to the principle of orthogonality. Naturally, we may derive this system of equations in its own independent way by formulating the gradient vector $\nabla\mathscr{E}$ in terms of the tap weights of the filter, and then solving for the tap-weight vector $\hat{\mathbf{w}}$ for which $\nabla\mathscr{E}$ is zero. Alternatively, we may derive the system of normal equations from the principle of orthogonality. We are going to pursue the latter (indirect) approach in this section, and leave the former (direct) approach to the interested reader as Problem 7.

The principle of orthogonality in Eq. (11.15) is formulated in terms of a set of tap inputs and the minimum estimation error $e_{\min}(i)$. Setting the tap weights in Eq. (11.4) to their least-squares values, we get

$$e_{\min}(i) = d(i) - \sum_{t=0}^{M-1} \hat{w}_t^* u(i-t) \tag{11.27}$$

where on the right-hand side we have purposely used t as the dummy summation index. Hence, substituting Eq. (11.27) in (11.15), and then rearranging terms, we get a system of M simultaneous equations:

$$\sum_{t=0}^{M-1} \hat{w}_t \sum_{i=M}^{N} u(i-k)u^*(i-t) = \sum_{i=M}^{N} u(i-k)d^*(i), \qquad k = 0, \ldots, M-1 \tag{11.28}$$

The two summations in Eq. (11.28) involving the index i represent time-averages, except for a scaling factor. They have the following interpretations:

1. The time average (over i) on the left-hand side of Eq. (11.28) represents the *time averaged autocorrelation function* of the tap inputs in the transversal filter of Fig. 11.2. In particular, we may write

$$\phi(t,k) = \sum_{i=M}^{N} u(i-k)u^*(i-t), \qquad 0 \le (t,k) \le M-1 \tag{11.29}$$

2. The time average (also over i) on the right-hand side of Eq. (11.28) represents the *cross-correlation* between the tap inputs and the desired response. In particular, we may write

$$z(-k) = \sum_{i=M}^{N} u(i-k)d^*(i), \qquad 0 \le k \le M-1 \tag{11.30}$$

Accordingly, we may rewrite the system of simultaneous equations (11.28) as follows:

$$\sum_{i=0}^{M-1} \hat{w}_i \phi(t, k) = z(-k), \qquad k = 0, 1, \ldots, M-1 \tag{11.31}$$

The system of equations (11.31) represents *the expanded system of the normal equations* for a linear least-squares filter.

Matrix Formulation of the Normal Equations

We may recast this system of equations in matrix form by first introducing the following definitions:

1. The M-by-M *time-averaged correlation matrix* of the tap inputs $u(i)$, $u(i-1)$, $\ldots$, $u(i-M+1)$:

$$\mathbf{\Phi} = \begin{bmatrix} \phi(0,0) & \phi(1,0) & \cdots & \phi(M-1,0) \\ \phi(0,1) & \phi(1,1) & \cdots & \phi(M-1,1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi(0,M-1) & \phi(1,M-1) & \cdots & \phi(M-1,M-1) \end{bmatrix} \tag{11.32}$$

2. The M-by-1 *time-averaged cross-correlation vector* between the tap inputs $u(i)$, $u(i-1), \ldots, u(i-M+1)$ and the desired response $d(i)$:

$$\mathbf{z} = [z(0), z(-1), \ldots, z(-M+1)]^T \tag{11.33}$$

3. The M-by-1 tap-weight vector of the least-squares filter:

$$\hat{\mathbf{w}} = [\hat{w}_0, \hat{w}_1, \ldots, \hat{w}_{M-1}]^T \tag{11.34}$$

Hence, in terms of these matrix definitions, we may now rewrite the system of M simultaneous equations (11.31) simply as

$$\mathbf{\Phi}\hat{\mathbf{w}} = \mathbf{z} \tag{11.35}$$

Equation (11.35) is *the matrix form of the normal equations for linear least-squares filters.*

Assuming that $\mathbf{\Phi}$ is nonsingular and therefore the inverse matrix $\mathbf{\Phi}^{-1}$ exists, we may solve Eq. (11.35) for the tap-weight vector of the linear least-squares filter:

$$\hat{\mathbf{w}} = \mathbf{\Phi}^{-1}\mathbf{z} \tag{11.36}$$

The condition for the existence of the inverse matrix $\mathbf{\Phi}^{-1}$ is discussed in Section 11.6.

Equation (11.36) is a very important result. In particular, it is the linear least-squares counterpart to the solution of the matrix form of the Wiener–Hopf equations (5.36). Basically, Eq. (11.36) states that the tap-weight vector $\hat{\mathbf{w}}$ of a linear least-squares filter is uniquely defined by the product of the inverse of the time-averaged correlation matrix $\mathbf{\Phi}$ of the tap inputs of the filter and the time-averaged cross-correlation vector $\mathbf{z}$ between the tap inputs and the desired response. Indeed, this equation is fundamental to the development of all recursive formulations of the linear least-squares filter, as pursued in subsequent chapters of the book.

Minimum Sum of Error Squares

Equation (11.26), derived in the preceding section, defines the minimum sum of error squares $\mathscr{E}_{\min}$. We now complete the evaluation of $\mathscr{E}_{\min}$, expressed as the difference between the energy $\mathscr{E}_d$ of the desired response and the energy $\mathscr{E}_{\text{est}}$ of its estimate. Usually, $\mathscr{E}_d$ is determined from the time series representing the desired response. To evaluate $\mathscr{E}_{\text{est}}$, we write

$$\begin{aligned}\mathscr{E}_{\text{est}} &= \sum_{i=M}^{N} |\hat{d}(i|\mathcal{U}_i)|^2 \\ &= \sum_{i=M}^{N} \sum_{t=0}^{M-1} \sum_{k=0}^{M-1} \hat{w}_t \hat{w}_k^* u(i-k)u^*(i-t) \\ &= \sum_{t=0}^{M-1} \sum_{k=0}^{M-1} \hat{w}_t \hat{w}_k^* \sum_{i=M}^{N} u(i-k)u^*(i-t)\end{aligned} \tag{11.37}$$

where, in the second line, we have made use of Eq. (11.18). The inner summation over time i in the final line of Eq. (11.37) represents the time-averaged autocorrelation function $\phi(t, k)$ [see Eq. (11.29)]. Hence, we may rewrite Eq. (11.37) as

$$\begin{aligned}\mathscr{E}_{\text{est}} &= \sum_{t=0}^{M-1} \sum_{k=0}^{M-1} \hat{w}_t^* \phi(t, k) \hat{w}_t \\ &= \hat{\mathbf{w}}^H \mathbf{\Phi} \hat{\mathbf{w}}\end{aligned} \tag{11.38}$$

where $\hat{\mathbf{w}}$ is the least-squares tap-weight vector and $\mathbf{\Phi}$ is the time-averaged correlation matrix of the tap inputs. We may further simplify the formula for $\mathscr{E}_{\text{est}}$ by noting that from the normal equations (11.35), the matrix product $\mathbf{\Phi}\hat{\mathbf{w}}$ equals the cross-correlation vector $\mathbf{z}$. Accordingly, we have

$$\begin{aligned}\mathscr{E}_{\text{est}} &= \hat{\mathbf{w}}^H \mathbf{z} \\ &= \mathbf{z}^H \hat{\mathbf{w}}\end{aligned} \tag{11.39}$$

Finally, substituting Eq. (11.39) in (11.26), and then using Eq. (11.36) for $\hat{\mathbf{w}}$, we get

$$\begin{aligned}\mathscr{E}_{\min} &= \mathscr{E}_d - \mathbf{z}^H\hat{\mathbf{w}} \\ &= \mathscr{E}_d - \mathbf{z}^H\boldsymbol{\Phi}^{-1}\mathbf{z}\end{aligned} \tag{11.40}$$

Equations (11.40) is the formula for the minimum sum of error squares, expressed in terms of three known quantities: the energy $\mathscr{E}_d$ of the desired response, the time-averaged correlation matrix $\boldsymbol{\Phi}$ of the tap inputs, and the time-averaged cross-correlation vector $\mathbf{z}$ between the tap inputs and the desired response.

11.6 TIME-AVERAGED CORRELATION MATRIX Φ

The time-averaged correlation matrix or simply the correlation matrix $\boldsymbol{\Phi}$ of the tap inputs is shown in its expanded form in Eq. (11.32), with the element $\phi(t, k)$ defined in Eq. (11.29). The index k in $\phi(t, k)$ refers to the row number in the matrix $\boldsymbol{\Phi}$, and t refers to the column number. Let the M-by-1 tap-input vector $\mathbf{u}(i)$ be defined by

$$\mathbf{u}(i) = [u(i), u(i-1), \ldots, u(i-M+1)]^T \tag{11.41}$$

Hence, we may use Eqs. (11.29) and (11.41) to redefine the correlation matrix $\boldsymbol{\Phi}$ as the time average of the outer product $\mathbf{u}(i)\mathbf{u}^H(i)$ over i as follows:

$$\boldsymbol{\Phi} = \sum_{i=M}^{N} \mathbf{u}(i)\mathbf{u}^H(i) \tag{11.42}$$

To restate what we said earlier under footnote 1, the summation in Eq. (11.42) should be divided by the scaling factor $(N - M + 1)$ for the correlation matrix $\boldsymbol{\Phi}$ to be a time average in precise terms. In the statistics literature, this scaled form of $\boldsymbol{\Phi}$ is referred to as the *sample correlation matrix*. In any event, on the basis of the definition given in Eq. (11.42), we may readily establish the following properties of the correlation matrix:

Property 1. *The correlation matrix $\boldsymbol{\Phi}$ is Hermitian; that is*

$$\boldsymbol{\Phi}^H = \boldsymbol{\Phi}$$

This property follows directly from Eq. (11.42).

Property 2. *The correlation matrix $\boldsymbol{\Phi}$ is nonnegative definite; that is,*

$$\mathbf{x}^H\boldsymbol{\Phi}\mathbf{x} \geq 0$$

for any M-by-1 vector $\mathbf{x}$.

Using the definition of Eq. (11.42), we may write

$$\begin{aligned}\mathbf{x}^H\mathbf{\Phi}\mathbf{x} &= \sum_{i=M}^{N} \mathbf{x}^H\mathbf{u}(i)\mathbf{u}^H(i)\mathbf{x} \\ &= \sum_{i=M}^{N} [\mathbf{x}^H\mathbf{u}(i)][\mathbf{x}^H\mathbf{u}(i)]^H \\ &= \sum_{i=M}^{N} |\mathbf{x}^H\mathbf{u}(i)|^2 \geq 0\end{aligned}$$

which proves Property 2. The fact that the correlation matrix $\mathbf{\Phi}$ is nonnegative definite means that its determinant and all principal minors are nonnegative. When the above condition is satisfied with the inequality sign, the determinant of $\mathbf{\Phi}$ and its principal minors are likewise nonzero. In the latter case, $\mathbf{\Phi}$ is nonsingular and the inverse $\mathbf{\Phi}^{-1}$ exists.

Property 3. *The eigenvalues of the correlation matrix $\mathbf{\Phi}$ are all real and nonnegative.*

The real requirement on the eigenvalues of $\mathbf{\Phi}$ follows from Property 1. The fact that all these eigenvalues are also nonnegative follows from Property 2.

Property 4. *The correlation matrix is the product of two rectangular Toeplitz matrices that are the Hermitian transpose of each other.*

The correlation matrix $\mathbf{\Phi}$ is, in general, non-Toeplitz, which is clearly seen by examining the expanded form of the correlation matrix given in Eq. (11.32). The elements on the main diagonal, $\phi(0, 0), \phi(1, 1), \ldots, \phi(M-1, M-1)$, have different values; this also applies to secondary diagonal above or below the main diagonal. However, the matrix $\mathbf{\Phi}$ has a special structure in the sense that it is the product of two Toeplitz rectangular matrices. To prove this property, we first use Eq. (11.42) to express the matrix $\mathbf{\Phi}$ as follows:

$$\mathbf{\Phi} = [\mathbf{u}(M), \mathbf{u}(M+1), \ldots, \mathbf{u}(N)] \begin{bmatrix} \mathbf{u}^H(M) \\ \mathbf{u}^H(M-1) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{u}^H(N) \end{bmatrix} \tag{11.43}$$

Next, for convenience of presentation, we introduce a *data matrix* $\mathbf{A}$, whose Hermitian transpose is defined by

$$\begin{aligned}\mathbf{A}^H = &\quad [\mathbf{u}(M), \quad \mathbf{u}(M+1), \quad \cdots, \mathbf{u}(N)] \\ &= \begin{bmatrix} u(M) & u(M+1) & \cdots & u(N) \\ u(M-1) & u(M) & \cdots & u(N-1) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ u(1) & u(2) & \cdots & u(N-M+1) \end{bmatrix}\end{aligned} \tag{11.44}$$

The expanded matrix on the right-hand side of Eq. (11.44) is recognized to be the matrix of input data for the covariance method of data windowing (see point 1 of Section 11.2). Thus, using the definition of Eq. (11.44), we may rewrite Eq. (11.43) in the compact form

$$\boldsymbol{\Phi} = \mathbf{A}^H\mathbf{A} \tag{11.45}$$

From the expanded form of the matrix given in the second line of Eq. (11.44), we see that $\mathbf{A}^H$ consists of an M- by-$(N - M + 1)$ *rectangular Toeplitz matrix*. The data matrix $\mathbf{A}$ itself is likewise an $(N - M + 1)$-by-M rectangular Toeplitz matrix. According to Eq. (11.45), therefore, the correlation matrix $\boldsymbol{\Phi}$ is the product of two rectangular Toeplitz matrices that are the Hermitian transpose of each other: this completes the proof of Property 4.

11.7 REFORMULATION OF THE NORMAL EQUATIONS IN TERMS OF DATA MATRICES

The system of normal equations for a least-squares transversal filter is given by Eq. (11.35) in terms of the correlation matrix $\boldsymbol{\Phi}$ and the cross-correlation vector $\mathbf{z}$. We may reformulate the normal equations in terms of data matrices by using Eq. (11.45) for the correlation matrix $\boldsymbol{\Phi}$ of the tap inputs, and a corresponding relation for the cross-correlation vector $\mathbf{z}$ between the tap inputs and the desired response. To do this, we introduce a *desired data vector* $\mathbf{d}$, consisting of the *desired response* $d(i)$ for values of i in the interval $[M, N]$; in particular, we define

$$\mathbf{d}^H = [d(M), d(M+1), \ldots, d(N)] \tag{11.46}$$

Note that we have purposely used Hermitian transposition rather than ordinary transposition in the definition of vector $\mathbf{d}$ to be consistent with the definition of the data matrix $\mathbf{A}$ in Eq. (11.44). With the definitions of Eqs. (11.44) and (11.46) at hand, we may now use Eqs. (11.30) and (11.33) to express the cross-correlation vector $\mathbf{z}$ as

$$\mathbf{z} = \mathbf{A}^H\mathbf{d} \tag{11.47}$$

Furthermore, we may use Eqs. (11.45) and (11.47) in (11.35), and so express the system of normal equations in terms of the data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$ as

$$\mathbf{A}^H\mathbf{A}\mathbf{w} = \mathbf{A}^H\mathbf{d}$$

Hence, the system of equations used in the minimization of the cost function $\mathscr{E}$ may be represented by $\mathbf{A}\mathbf{w} = \mathbf{d}$. Furthermore, assuming that the inverse matrix $(\mathbf{A}^H\mathbf{A})^{-1}$ exists, we may solve this system of equations by expressing the tap-weight vector $\hat{\mathbf{w}}$ as

$$\hat{\mathbf{w}} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{d} \tag{11.48}$$

We may complete the reformulation of our results for the linear least-squares problem in terms of the data matrices $\mathbf{A}$ and $\mathbf{d}$ by using (1) the definitions of Eqs. (11.45) and (11.47) in (11.40), and (2) the definitions of Eq. (11.46) in (11.23). By so doing, we may rewrite the formula for the minimum sum of error squares as

$$\mathscr{E}_{\min} = \mathbf{d}^H\mathbf{d} - \mathbf{d}^H\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{d} \tag{11.49}$$

Although this formula looks somewhat cumbersome, its nice feature is that it is expressed explicitly in terms of the data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$.

Projection Operator

Equation (11.48) defines the least-squares tap-weight vector $\hat{\mathbf{w}}$ in terms of the data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$. The least-squares estimate of $\mathbf{d}$ is therefore given by

$$\begin{aligned}\hat{\mathbf{d}} &= \mathbf{A}\hat{\mathbf{w}} \\ &= \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{d}\end{aligned} \tag{11.50}$$

Accordingly, we may view the multiple matrix product $\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$ as a *projection operator* onto the linear space spanned by the columns of the data matrix $\mathbf{A}$, which is the same space $\mathscr{U}_i$ mentioned previously for $i = N$. Denoting this projection operator by $\mathbf{P}$, we may thus write

$$\mathbf{P} = \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H \tag{11.51}$$

The matrix difference

$$\mathbf{I} - \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H = \mathbf{I} - \mathbf{P}$$

is the *orthogonal complement projector*. Note that both the projection operator and its complement are uniquely determined by the data matrix A. The projection operator, $\mathbf{P}$, applied to the desired data vector $\mathbf{d}$, yields the corresponding estimate $\hat{\mathbf{d}}$. On the other hand, the orthogonal complement projector, $\mathbf{I} - \mathbf{P}$, applied to the desired data vector $\mathbf{d}$, yields the estimation error vector $\mathbf{e}_{min} = \mathbf{d} - \hat{\mathbf{d}}$. Figure 11.3 illustrates the functions of the projection operator and the orthogonal complement projector as described herein.

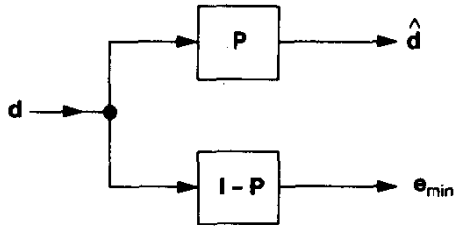

Figure 11.3 Projection operator **P** and orthogonal complement projector **I** − **P**.

Example 1

Consider the example of a linear least-squares filter with two taps (i.e., $M = 2$) and a *real-valued* input time series consisting of four samples (i.e., $N = 4$), hence $N - M + 1 = 3$. The input data matrix **A** and the desired data vector **d** have the following values:

$$\mathbf{A} = \begin{bmatrix} u(2) & u(1) \\ u(3) & u(2) \\ u(4) & u(3) \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 3 \\ 1 & 2 \\ -1 & 1 \end{bmatrix}$$

$$\mathbf{d} = \begin{bmatrix} d(2) \\ d(3) \\ d(4) \end{bmatrix}$$

$$= \begin{bmatrix} 2 \\ 1 \\ 1/34 \end{bmatrix}$$

The purpose of this example is to evaluate the projection operator and the orthogonal complement projector, and use them to illustrate the principle of orthogonality.

The use of Eq. (11.51), reformulated for real data, yields the value of the projection operator **P** as

$$\mathbf{P} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$$

$$= \frac{1}{35}\begin{bmatrix} 26 & 15 & -2 \\ 15 & 10 & 5 \\ -3 & 5 & 34 \end{bmatrix}$$

The corresponding value of the orthogonal complement projector is

$$\mathbf{I} - \mathbf{P} = \frac{1}{35}\begin{bmatrix} 9 & -15 & 3 \\ -15 & 25 & -5 \\ -3 & -5 & 1 \end{bmatrix}$$

Accordingly, the estimate of the desired data vector and the estimation error vector have the following values, respectively:

$$\hat{\mathbf{d}} = \mathbf{P}\mathbf{d} = \begin{bmatrix} 1.91 \\ 1.15 \\ 0 \end{bmatrix}$$

$$\mathbf{e}_{\min} = (\mathbf{I} - \mathbf{P})\mathbf{d} = \begin{bmatrix} 0.09 \\ -0.15 \\ 0.03 \end{bmatrix}$$

Figure 11.4 depicts three-dimensional geometric representations of the vectors $\hat{\mathbf{d}}$ and $\mathbf{e}_{\min}$. This figure clearly shows that these two vectors are *normal* (i.e., *perpendicular*) to each other in accordance with the corollary to the principle of orthogonality, hence the terminology "normal" equations. This condition is the geometric portrayal of the fact that in a linear least-squares filter the inner product $\mathbf{e}_{\min}^H\mathbf{d}$ is zero. Figure 11.4 also depicts the desired data vector $\mathbf{d}$ as the "vector sum" of the estimate $\hat{\mathbf{d}}$ and the error $\mathbf{e}_{\min}$. Note also that the vector $\mathbf{e}_{\min}$ is orthogonal to span($\mathbf{A}$), defined as the set of all linear combinations of the column vectors of the data matrix $\mathbf{A}$. The estimate $\hat{\mathbf{d}}$ is just one vector in span($\mathbf{A}$).

Uniqueness Theorem

The linear least-squares problem of minimizing the sum of error squares, $\mathscr{E}(n)$, always has a solution. That is, for given values of the data matrix $\mathbf{A}$ and the desired data vector $\hat{\mathbf{d}}$, we can always find a vector $\hat{\mathbf{w}}$ that satisfies the normal equations. It is therefore important that we know *if* and *when* the solution is *unique*. This requirement is covered by the following *uniqueness theorem* (Stewart, 1973):

The least-squares estimate $\hat{\mathbf{w}}$ is unique if and only if the nullity of the data matrix A equals zero.

Let $\mathbf{A}$ be a K-by-M matrix; in the case of the data matrix $\mathbf{A}$ defined in Eq. (11.44), we have $K = N - M + 1$. We define the *null space* of matrix $\mathbf{A}$, written as $\mathcal{N}(\mathbf{A})$, as the space of all vectors $\mathbf{x}$ such that $\mathbf{A}\mathbf{x} = \mathbf{0}$. We define the *nullity* of matrix $\mathbf{A}$, written as null($\mathbf{A}$), as the dimension of the null space $\mathcal{N}(\mathbf{A})$. In general, we find that

$$\text{null}(\mathbf{A}) \neq \text{null}(\mathbf{A}^H).$$

In light of the uniqueness theorem, which is intuitively satisfying, we may expect a unique solution to the linear least-squares problem *only* when the data matrix $\mathbf{A}$ has *linearly independent columns;* that is, when the data matrix $\mathbf{A}$ is of *full column rank*. This implies that the matrix $\mathbf{A}$ has at least as many rows as columns; that is, $(N - M + 1) \geq M$. This latter condition means that the system of equations represented by $\mathbf{A}\mathbf{w} = \mathbf{d}$ used in the minimization is *overdetermined*, in that it has more equations than unknowns. Thus,

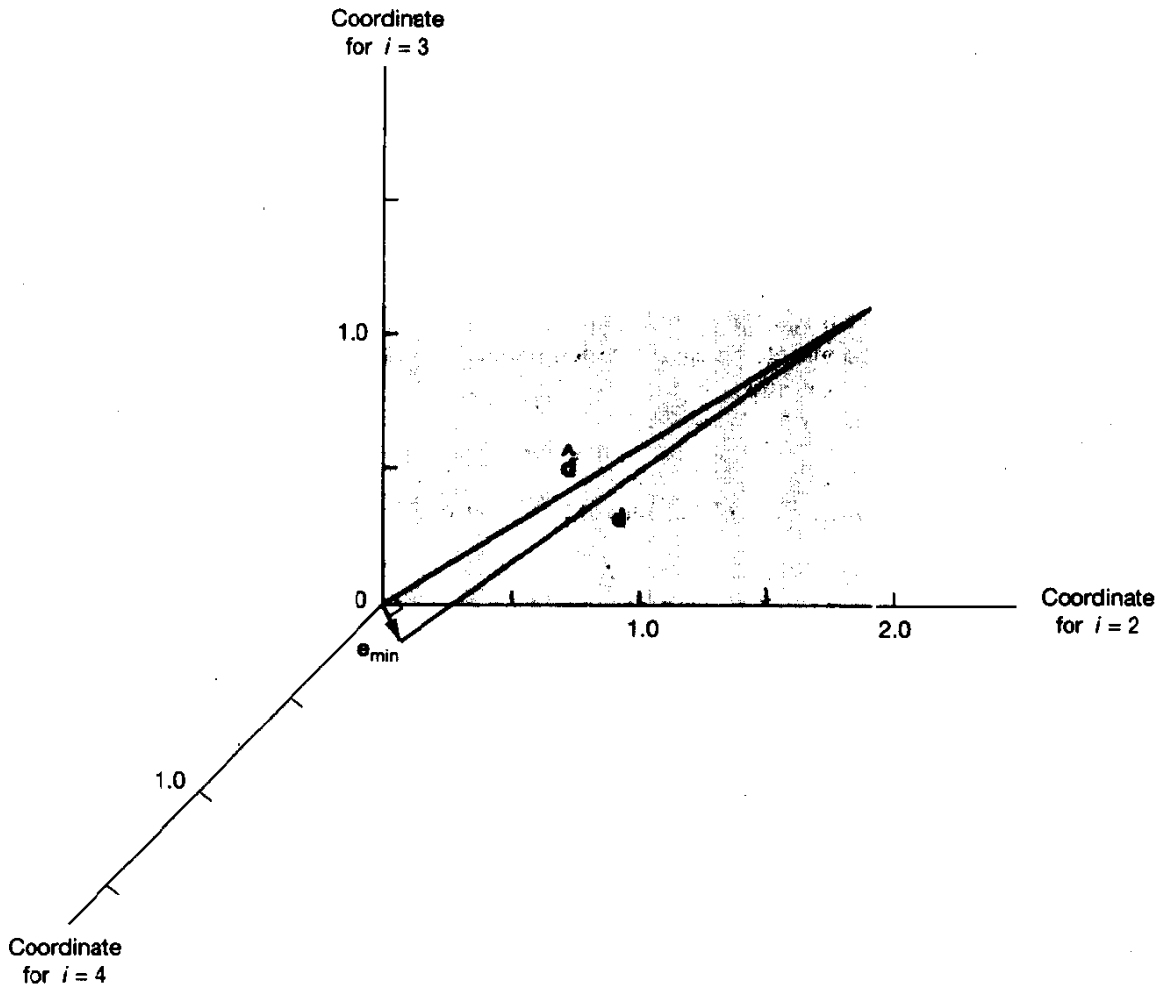

Figure 11.4 Three-dimensional geometric interpretations of vectors $\mathbf{d}$, $\hat{\mathbf{d}}$, and $\mathbf{e}_{min}$.

provided that the data matrix $\mathbf{A}$ is of full column rank, the M-by-M matrix $\mathbf{A}^H\mathbf{A}$ is *non-singular*, and the least-squares estimate has the unique value given in Eq. (11.48).

When, however, the matrix $\mathbf{A}$ has *linearly dependent columns*, that is, it is *rank deficient*, the nullity of the matrix $\mathbf{A}$ is nonzero, and the result is that an infinite number of solutions can be found for minimizing the sum of error squares. In such a situation, the linear least-squares problem becomes quite involved, in that we now have the new problem of deciding which particular solution to adopt. We defer discussion of this issue to the latter part of the chapter. In the meantime, we assume that the data matrix $\mathbf{A}$ is of full column rank, so that the least-squares estimate $\hat{\mathbf{w}}$ has the unique value defined by Eq. (11.48).

11.8 PROPERTIES OF LEAST-SQUARES ESTIMATES

The method of least squares has a strong intuitive feel that is reinforced by several outstanding properties of the method, assuming that the data matrix **A** is known with *no* uncertainty. These properties, four in number, are described next (Miller, 1974; Goodwin and Payne, 1977).

Property 1. *The least-squares estimate* $\hat{\mathbf{w}}$ is unbiased, provided that the measurement error process $e_o(i)$ *has zero mean.*

From the multiple linear regression model of Fig. 11.1, we have [using the definitions of Eqs. (11.44) and (11.46)]

$$\mathbf{d} = \mathbf{A}\mathbf{w}_o + \boldsymbol{\epsilon}_o \tag{11.52}$$

Hence, substituting Eq. (11.52) into (11.48), we may express the least-squares estimate $\hat{\mathbf{w}}$ as

$$\begin{aligned}\hat{\mathbf{w}} &= (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{A}\mathbf{w}_o + (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\boldsymbol{\epsilon}_o \\ &= \mathbf{w}_o + (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\boldsymbol{\epsilon}_o\end{aligned} \tag{11.53}$$

The matrix product $(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$ is a known quantity, since the data matrix **A** is completely defined by the set of given observations $u(1), u(2), \ldots, u(N)$; see Eq. (11.44). Hence, if the measurement error process $e_o(i)$ or, equivalently, the measurement error vector $\boldsymbol{\epsilon}_o$ has zero mean, we find by taking the expectation of both sides of Eq. (11.53) that the estimate $\hat{\mathbf{w}}$ is *unbiased*; that is,

$$E[\hat{\mathbf{w}}] = \mathbf{w}_o \tag{11.54}$$

Property 2. *When the measurement error process* $e_o(i)$ *is white with zero mean and variance* σ^2, *the covariance matrix of the least-squares estimate* $\hat{\mathbf{w}}$ equals $\sigma^2\boldsymbol{\Phi}^{-1}$.

Using the relation of Eq. (11.53), we find that the covariance matrix of the least-squares estimate $\hat{\mathbf{w}}$ equals

$$\begin{aligned}\text{cov}[\hat{\mathbf{w}}] &= E[(\hat{\mathbf{w}} - \mathbf{w}_o)(\hat{\mathbf{w}} - \mathbf{w}_o)^H] \\ &= E[(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}] \\ &= (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H E[\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H]\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\end{aligned} \tag{11.55}$$

With the measurement error process $e_o(i)$ assumed to be white with zero mean and variance σ^2, we have

$$E[\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H] = \sigma^2\mathbf{I} \tag{11.56}$$

where $\mathbf{I}$ is the identity matrix. Hence, Eq. (11.55) reduces to

$$\begin{aligned}\text{cov}[\hat{\mathbf{w}}] &= \sigma^2(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1} \\ &= \sigma^2(\mathbf{A}^H\mathbf{A})^{-1} \\ &= \sigma^2\boldsymbol{\Phi}^{-1}\end{aligned} \tag{11.57}$$

which proves Property 2.

Property 3. *When the measurement error process $e_o(i)$ is white with zero mean, the least-squares estimate $\hat{\mathbf{w}}$ is the best linear unbiased estimate.*

Consider any linear unbiased estimator $\tilde{\mathbf{w}}$ that is defined by

$$\tilde{\mathbf{w}} = \mathbf{B}\mathbf{d} \tag{11.58}$$

where $\mathbf{B}$ is an M-by-$(N - M + 1)$ matrix. Substituting Eq. (11.52) into (11.58), we get

$$\tilde{\mathbf{w}} = \mathbf{B}\mathbf{A}\mathbf{w}_o + \mathbf{B}\boldsymbol{\epsilon}_o \tag{11.59}$$

With the measurement error vector $\boldsymbol{\epsilon}_o$ having zero mean in accordance with Property 1, we find that the expected value of $\tilde{\mathbf{w}}$ equals

$$E[\tilde{\mathbf{w}}] = \mathbf{B}\mathbf{A}\mathbf{w}_o$$

For the linear estimator $\tilde{\mathbf{w}}$ to be unbiased, we therefore require that the matrix $\mathbf{B}$ satisfy the condition

$$\mathbf{B}\mathbf{A} = \mathbf{I}$$

Accordingly, we may rewrite Eq. (11.59) as follows:

$$\tilde{\mathbf{w}} = \mathbf{w}_o + \mathbf{B}\boldsymbol{\epsilon}_o$$

The covariance matrix of $\tilde{\mathbf{w}}$ equals

$$\begin{aligned}\text{cov}[\tilde{\mathbf{w}}] &= E[(\tilde{\mathbf{w}} - \mathbf{w}_o)(\tilde{\mathbf{w}} - \mathbf{w}_o)^H] \\ &= E[\mathbf{B}\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H\mathbf{B}^H] \\ &= \sigma^2\mathbf{B}\mathbf{B}^H\end{aligned} \tag{11.60}$$

Here, we have made use of Eq. (11.56), which describes the assumption that the elements of the measurement error vector $\boldsymbol{\epsilon}_o$ are uncorrelated and have a common variance σ^2; that is, the measurement error process $e_o(i)$ is white. We next define a new matrix $\boldsymbol{\Psi}$ in terms of $\mathbf{B}$ as

$$\boldsymbol{\Psi} = \mathbf{B} - (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H \tag{11.61}$$

Now we form the matrix product $\Psi\Psi^H$ and note that $\mathbf{BA} = \mathbf{I}$:

$$\begin{aligned}\Psi\Psi^H &= [\mathbf{B} - (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H][\mathbf{B}^H - \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}] \\ &= \mathbf{B}\mathbf{B}^H - \mathbf{B}\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1} - (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{B}^H + (\mathbf{A}^H\mathbf{A})^{-1} \\ &= \mathbf{B}\mathbf{B}^H - (\mathbf{A}^H\mathbf{A})^{-1}\end{aligned}$$

Since the diagonal elements of $\Psi\Psi^H$ are always nonnegative, we may use this relation to write

$$\sigma^2 \operatorname{diag}[\mathbf{B}\mathbf{B}^H] \geq \sigma^2 \operatorname{diag}[(\mathbf{A}^H\mathbf{A})^{-1}] \tag{11.62}$$

The term $\sigma^2\mathbf{B}\mathbf{B}^H$ equals the covariance matrix of the linear estimate $\tilde{\mathbf{w}}$, as in Eq. (11.60). From Property 2, we also know that the term $\sigma^2(\mathbf{A}^H\mathbf{A})^{-1}$ equals the covariance matrix of the least-squares estimate $\hat{\mathbf{w}}$. Thus, Eq. (11.62) shows that within the class of linear unbiased estimates the least-squares estimate $\hat{\mathbf{w}}$ is the "best" estimate of the unknown parameter vector $\mathbf{w}_o$ of the multiple linear regression model, in the sense that each element of $\hat{\mathbf{w}}$ has the smallest possible variance. Accordingly, when the measurement error process e_o contained in this model is white with zero mean, the least-squares estimate $\hat{\mathbf{w}}$ is the *b*est *l*inear *u*nbiased *e*stimate (BLUE).

Thus far we have not made any assumption about the statistical distribution of the measurement error process $e_o(i)$ other than that it is a zero mean white-noise process. By making the further assumption that the process $e_o(i)$ is *Gaussian* distributed, we obtain a stronger result on the optimality of the linear least-squares estimate, as discussed next.

Property 4. *When the measurement error process $e_o(i)$ is white and Gaussian, with zero mean, the least-squares estimate $\hat{\mathbf{w}}$ achieves the Cramér–Rao lower bound for unbiased estimates.*

Let $f_E(\boldsymbol{\epsilon}_o)$ denote the joint probability density function of the measurement error vector $\boldsymbol{\epsilon}_o$. Let $\hat{\mathbf{w}}$ denote any unbiased estimate of the unknown parameter vector $\hat{\mathbf{w}}_o$ of the multiple linear regression model. Then the covariance matrix of $\hat{\mathbf{w}}$ satisfies the inequality

$$\operatorname{cov}[\hat{\mathbf{w}}] \geq \mathbf{J}^{-1} \tag{11.63}$$

where

$$\operatorname{cov}[\hat{\mathbf{w}}] = E[(\hat{\mathbf{w}} - \mathbf{w}_o)(\hat{\mathbf{w}} - \mathbf{w}_o)^H] \tag{11.64}$$

The matrix $\mathbf{J}$ is called *Fisher's information matrix*; it is defined by[2]

$$\mathbf{J} = E\left[\left(\frac{\partial l}{\partial \mathbf{w}_o^*}\right)\left(\frac{\partial l}{\partial \mathbf{w}_o^T}\right)\right] \tag{11.65}$$

where l is the *log-likelihood function*, that is, the natural logarithm of the joint probability density of $\boldsymbol{\epsilon}_o$, as shown by

$$l = \ln f_E(\boldsymbol{\epsilon}_o) \tag{11.66}$$

[2]Fisher's information matrix is discussed in Appendix D for the case of real parameters.

Since the measurement error process $e_o(n)$ is white, the elements of the vector $\boldsymbol{\epsilon}_o$ are uncorrelated. Furthermore, since the process $e_o(n)$ is Gaussian, the elements of $\boldsymbol{\epsilon}_o$ are statistically independent. With $e_o(i)$ assumed to be complex with a mean of zero and variance σ^2, we have (see Section 2.11)

$$f_E(\boldsymbol{\epsilon}_o) = \frac{1}{(\pi\sigma^2)^{(N-M+1)}} \exp\left[-\frac{1}{\sigma^2}\sum_{i=M}^{N} |e_o(i)|^2\right] \tag{11.67}$$

The log-likelihood function is therefore

$$\begin{aligned} l &= F - \frac{1}{\sigma^2}\sum_{i=M}^{N} |e_o(i)|^2 \\ &= F - \frac{1}{\sigma^2}\boldsymbol{\epsilon}_o^H\boldsymbol{\epsilon}_o \end{aligned} \tag{11.68}$$

where F is a constant defined by

$$F = -(N - M + 1)\ln(\pi\sigma^2)$$

From Eq. (11.52), we have

$$\boldsymbol{\epsilon}_o = \mathbf{d} - \mathbf{A}\mathbf{w}_o$$

Using this relation in Eq. (11.68), we may rewrite l in terms of $\mathbf{w}_o$ as

$$l = F - \frac{1}{\sigma^2}\mathbf{d}^H\mathbf{d} + \frac{1}{\sigma^2}\mathbf{w}_o^H\mathbf{A}^H\mathbf{d} + \frac{1}{\sigma^2}\mathbf{d}^H\mathbf{A}\mathbf{w}_o - \frac{1}{\sigma^2}\mathbf{w}_o^H\mathbf{A}^H\mathbf{A}\mathbf{w}_o \tag{11.69}$$

Differentiating the real-valued log-likelihood function l with respect to the complex-valued unknown parameter vector $\mathbf{w}_o$ in accordance with the notation described in Appendix B, we get

$$\begin{aligned} \frac{\partial l}{\partial \mathbf{w}_o^*} &= \frac{1}{\sigma^2}\mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{w}_o) \\ &= \frac{1}{\sigma^2}\mathbf{A}^H\boldsymbol{\epsilon}_o \end{aligned} \tag{11.70}$$

Thus, substituting Eq. (11.70) into (11.65) yields Fisher's information matrix for the problem at hand as

$$\begin{aligned} \mathbf{J} &= \frac{1}{\sigma^4}E[\mathbf{A}^H\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H\mathbf{A}] \\ &= \frac{1}{\sigma^4}\mathbf{A}^H E[\boldsymbol{\epsilon}_o\boldsymbol{\epsilon}_o^H]\mathbf{A} \\ &= \frac{1}{\sigma^2}\mathbf{A}^H\mathbf{A} \\ &= \frac{1}{\sigma^2}\boldsymbol{\Phi} \end{aligned} \tag{11.71}$$

where, in the third line, we have made use of Eq. (11.56) describing the assumption that the measurement error process $e_o(i)$ is white with zero mean and variance σ^2. Accordingly, the use of Eq. (11.63) shows that the covariance matrix of the unbiased estimate $\tilde{\mathbf{w}}$ satisfies the inequality

$$\text{cov}[\tilde{\mathbf{w}}] \geq \sigma^2 \boldsymbol{\Phi}^{-1} \tag{11.72}$$

However, from Property 2, we know that $\sigma^2\boldsymbol{\Phi}^{-1}$ equals the covariance matrix of the least-squares estimate $\hat{\mathbf{w}}$. Accordingly, $\hat{\mathbf{w}}$ achieves the Cramér–Rao lower bound. Moreover, using Property 1, we conclude that when the measurement error process $e_o(i)$ is a zero-mean white Gaussian noise process, the least-squares estimate $\hat{\mathbf{w}}$ is a *m*inimum *v*ariance *u*nbiased *e*stimate (MVUE).

11.9 PARAMETRIC SPECTRUM ESTIMATION

The method of least squares is particularly well suited for solving *parametric spectrum estimation* problems. In this section we study this important application of the method of least squares. We first consider the case of *autoregressive (AR) spectrum estimation*, assuming the use of an AR model of *known order*. From the discussion of linear prediction presented in Chapter 6, we know that there is a one-to-one correspondence between the coefficients of a prediction-error filter and those of an AR model of similar order. Next, we consider the case of *minimum variance distortionless response (MVDR) spectrum estimation*. In this second case, we have a constrained optimization problem to solve.

AR Spectrum Estimation

The specific estimation procedure described herein relies on the combined use of *forward and backward linear prediction (FBLP)*.[3] Since the method of least squares is basically a *block* estimation procedure, we may therefore view the FBLP algorithm as an alternative to the Burg algorithm (described in Section 6.15) for solving AR modeling problems. There are, however, three basic differences between the FBLP and the Burg algorithms:

1. The FBLP algorithm estimates the coefficients of a *transversal-equivalent model* for the input data, whereas the Burg algorithm estimates the reflection coefficients of a *lattice-equivalent model*.
2. In the method of least squares, and therefore the FBLP algorithm, no assumptions are made concerning the statistics of the input data. The Burg algorithm, on the other hand, exploits the decoupling property of a multistage lattice predictor, which, in turn, assumes wide-sense stationarity of the input data. Accordingly, the

[3]The first application of the FBLP method to the design of a linear predictor that has a transversal filter structure, in accordance with the method of least squares, was developed independently by Ulrych and Clayton (1976) and Nuttall (1976).

FBLP algorithm does not suffer from some of the anomalies that are known to arise in the application of the Burg algorithm.[4]

3. The Burg algorithm yields a minimum-phase solution in the sense that the reflection coefficients of the equivalent lattice predictor have a magnitude less than or equal to unity. The FBLP algorithm, on the other hand, does *not* guarantee such a solution. In spectrum estimation, however, the lack of a minimum-phase solution is of no particular concern.

Consider then the *forward linear predictor*, shown in Fig. 11.5(a). The tap weights of the predictor are denoted by $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_M$ and the tap inputs by $u(i-1), u(i-2), \ldots, u(i-M)$, respectively. The forward prediction error, denoted by $f_M(i)$, equals

$$f_M(i) = u(i) - \sum_{k=1}^{M} \hat{w}_k^* u(i-k) \tag{11.73}$$

The first term, $u(i)$, represents the desired response. The convolution sum, constituting the second term, represents the predictor output; it consists of the sum of scalar inner products. Using matrix notation, we may also express the forward prediction error as

$$f_M(i) = u(i) - \mathbf{w}^H \mathbf{u}(i-1) \tag{11.74}$$

where $\hat{\mathbf{w}}$ is the M-by-1 tap-weight vector of the predictor:

$$\mathbf{w} = [\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_M]^T$$

and $\mathbf{u}(i-1)$ is the corresponding tap-input vector:

$$\mathbf{u}(i-1) = [u(i-1), u(i-2), \ldots, u(i-M)]^T$$

Consider next Fig. 11.5(b), which depicts the reconfiguration of the predictor so that it performs backward linear prediction. We have *purposely* retained $\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_M$ as the tap weights of the predictor. The change in the format of the tap inputs is inspired by the discussion presented in Section 6.2 on backward linear prediction and its relation to forward linear prediction for the case of wide-sense stationary inputs. In particular, the tap inputs in the predictor of Fig. 11.5(b) differ from those of the forward linear predictor of Fig. 11.5(a) in two respects:

1. The tap inputs in Fig. 11.5(b) are *time reversed*, in that they appear from right to left whereas in Fig. 11.5(a) they appear from left to right.

[4]For example, when the Burg algorithm is used to estimate the frequency of an unknown sine wave in additive noise, under certain conditions a phenomenon commonly referred to as *spectral line splitting* may occur. This phenomenon refers to the occurrence of two (or more) closely spaced spectral peaks where there should only be a single peak; for a discussion of spectral line splitting, see Marple (1987), Kay (1988), and Haykin (1989); the original reference is Fougere et al. (1976). This anomaly, however, does not arise in the application of the FBLP algorithm.

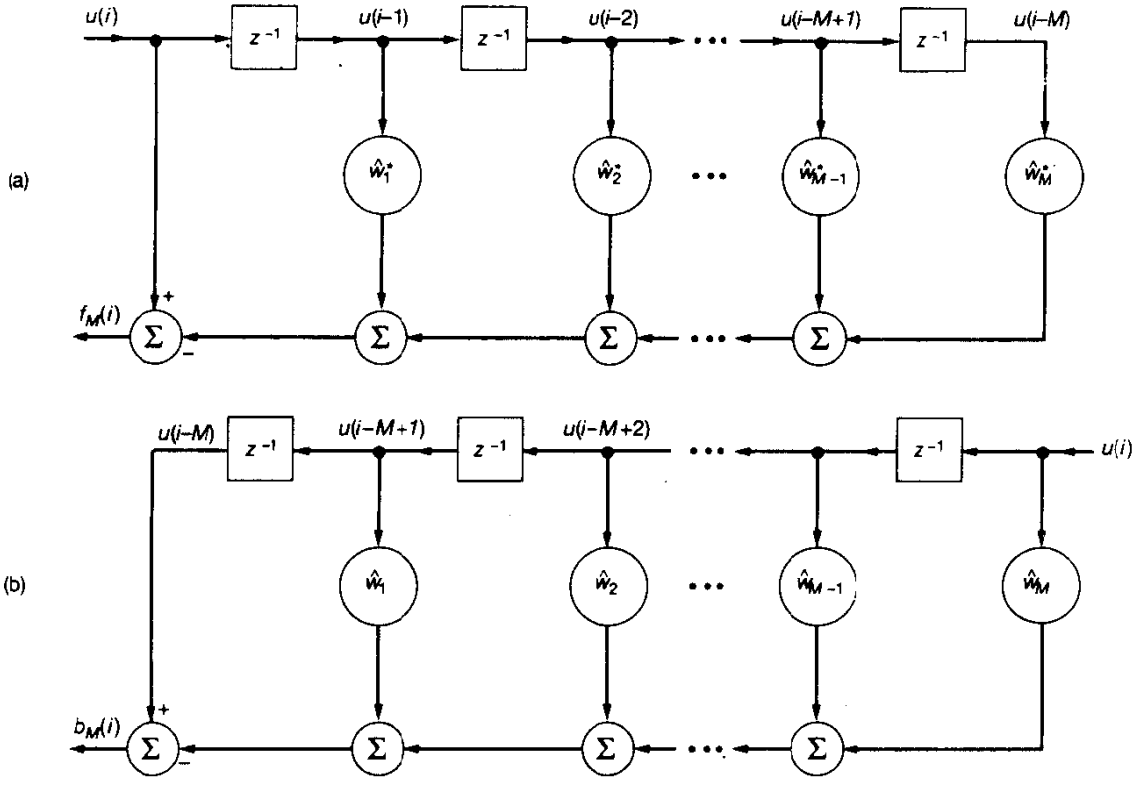

Figure 11.5 (a) Forward linear predictor; (b) reconfiguration of the predictor so as to perform backward linear prediction.

2. With $u(i)$, $u(i-1)$, . . . , $u(i-M+1)$ used as tap inputs, the structure of Fig. 11.5(b) produces a linear prediction of $u(i-M)$. In other words, it performs backward linear prediction. Denoting the backward prediction error by $b_M(i)$, we may thus express it as

$$b_M(i) = u(i-M) - \sum_{k=1}^{M} \hat{w}_k u(i-M+k) \tag{11.75}$$

where the first term represents the desired response and the second term is the predictor output. Equivalently, in terms of matrix notation, we may write

$$b_M(i) = u(i-M) - \mathbf{u}^{BT}(i)\hat{\mathbf{w}} \tag{11.76}$$

where $\mathbf{u}^B(i)$ is the *time-reversed tap-input vector*:

$$\mathbf{u}^{BT}(i) = [u(i-M+1), \ldots, u(i-1), u(i)]$$

Let $\mathscr{E}_M$ denote the *minimum value of the forward-backward prediction-error energy*. In accordance with the method of least squares, we may therefore write

$$\mathscr{E}_M = \sum_{i=M+1}^{N} [|f_M(i)|^2 + |b_M(i)|^2] \tag{11.77}$$

where the subscript M signifies the order of the predictor or that of the AR model. The lower limit on the time index i equals $M + 1$ so as to ensure that the forward and backward prediction errors are formed only when all the tap inputs of interest assume nonzero values. In particular, we may make two observations:

1. The variable $u(i - M)$, representing the last tap input in the forward prediction of Fig. 11.5(a), assumes a nonzero value for the first time when $i = M + 1$.
2. The variable $u(i - M)$, playing the role of desired response in the backward predictor of Fig. 11.5(b), also assumes a nonzero value for the first time when $i = M + 1$.

Thus, by choosing $(M + 1)$ as the lower limit on i and N as the upper limit, as in Eq. (11.77), we make no assumptions about the data outside the interval $[1, N]$, as required by the covariance method.

Let $\mathbf{A}$ denote the $2(N - M)$-by-M *data matrix*, whose Hermitian transpose is defined by

$$\mathbf{A}^H = \begin{bmatrix} u(M) & \cdots & u(N-1) & u^*(2) & \cdots & u^*(N-M+1) \\ u(M-1) & \cdots & u(N-2) & u^*(3) & \cdots & u^*(N-M+2) \\ \vdots & & \vdots & \vdots & & \vdots \\ u(1) & \cdots & u(N-M) & u^*(M+1) & \cdots & u^*(N) \end{bmatrix}$$

forward half | backward half (11.78)

The elements constituting the left half of matrix $\mathbf{A}^H$ represent the various sets of tap inputs used to make a total of $(N - M)$ *forward* linear predictions. The complex-conjugated elements constituting the right half of matrix $\mathbf{A}^H$ represent the corresponding sets of tap inputs used to make a total of $(N - M)$ *backward* linear predictions. Note that as we move from one column to the next in the forward or backward half in Eq. (11.78), we drop a sample, add a new one, and reorder the samples.

Let $\mathbf{d}$ denote the $2(N - M)$-by-1 *desired data vector*, defined in a manner corresponding to that shown in Eq. (11.78):

$$\mathbf{d}^H = \underbrace{[u(M+1), \ldots, u(N)}_{\text{forward half}}, \underbrace{u^*(1), \ldots, u^*(N-M)]}_{\text{backward half}} \tag{11.79}$$

Each element in the left half of the vector $\mathbf{d}^H$ represents a desired response for forward linear prediction. Each complex-conjugated element in the right half represents a desired response for backward linear prediction.

The FBLP method is a product of the method of least squares; it is therefore described by the system of normal equations [see Eq. (11.48)]

$$\mathbf{A}^H\mathbf{A}\hat{\mathbf{w}} = \mathbf{A}^H\mathbf{d} \tag{11.80}$$

The resulting minimum value of the forward-backward prediction error energy equals [see Eq. (11.49)]

$$\mathscr{E}_{\min} = \mathbf{d}^H\mathbf{d} - \mathbf{d}^H\mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{d} \tag{11.81}$$

The data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$ are defined by Eqs. (11.78) and (11.79), respectively.

We may combine Eqs. (11.80) and (11.81) into a single matrix relation, as shown by

$$\begin{bmatrix} \mathbf{d}^H\mathbf{d} & \mathbf{d}^H\mathbf{A} \\ \mathbf{A}^H\mathbf{d} & \mathbf{A}^H\mathbf{A} \end{bmatrix} \begin{bmatrix} 1 \\ -\hat{\mathbf{w}} \end{bmatrix} = \begin{bmatrix} \mathscr{E}_{\min} \\ \mathbf{0} \end{bmatrix} \tag{11.82}$$

where $\mathbf{0}$ is the M-by-1 null vector. Equation (11.82) is the matrix form of the *augmented normal equations for FBLP*. Define the $(M + 1)$-by-$(M + 1)$ *augmented correlation matrix*:

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{d}^H\mathbf{d} & \mathbf{d}^H\mathbf{A} \\ \mathbf{A}^H\mathbf{d} & \mathbf{A}^H\mathbf{A} \end{bmatrix} \tag{11.83}$$

The $\mathbf{\Phi}$ in Eq. (11.83) is an $(M + 1)$-by-$(M + 1)$ matrix; it is *not* to be confused with the $\mathbf{\Phi}$ in Eq. (11.45) that is an M-by-M matrix. Define the $(M + 1)$-by-1 tap-weight vector of the *prediction-error filter of order M*:

$$\hat{\mathbf{a}} = \begin{bmatrix} 1 \\ -\hat{\mathbf{w}} \end{bmatrix} \tag{11.84}$$

Figure 11.6 shows the transversal structure of the prediction-error filter, where $a_0, a_1, \ldots, a_M$ denote the tap weights[5] and $a_0 = 1$. Then

$$\mathbf{\Phi}\hat{\mathbf{a}} = \begin{bmatrix} \mathscr{E}_{\min} \\ \mathbf{0} \end{bmatrix} \tag{11.85}$$

The augmented correlation matrix $\mathbf{\Phi}$ is *Hermitian persymmetric*; that is, the individual elements of the matrix $\mathbf{\Phi}$ satisfy two conditions:

$$\phi(k, i) = \phi^*(i, k) \qquad 0 \le (i, k) \le M \tag{11.86}$$

$$\phi(M - k, M - i) = \phi^*(k, i), \qquad 0 \le (i, k) \le M \tag{11.87}$$

[5]The subscripts assigned to the tap weights in the prediction-error filter of Fig. 11.6 do not include a direct reference to the prediction order M, unlike the terminology used in Chapter 6. The reason for this simplification is that, in the material presented here, there is no order update to be considered.

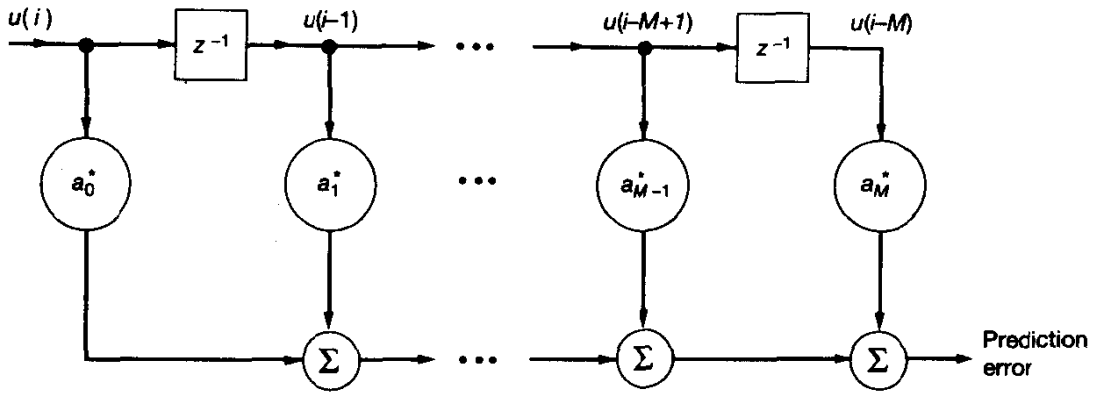

Figure 11.6 Forward prediction-error filter.

The property described in Eq. (11.87) is unique to a correlation matrix that is obtained by time averaging of the input data in the forward as well as backward direction; see the data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$ defined in Eqs. (11.78) and (11.79), respectively. The matrix $\mathbf{\Phi}$ has another property: it is composed of the sum of two Toeplitz matrix products. The special Toeplitz structure of the matrix $\mathbf{\Phi}$ has been exploited in the development of fast recursive algorithms[6] for the efficient solution of the augmented normal equations (11.85).

Starting with the time series $u(i)$, $1 \leq i \leq N$, the FBLP algorithm is used to compute the tap-weight vector $\hat{\mathbf{w}}$ of a forward linear predictor or, equivalently, the tap-weight vector $\hat{\mathbf{a}}$ of the corresponding prediction-error filter. The vector $\hat{\mathbf{a}}$ represents an estimate of the coefficient vector of an *autoregressive (AR) model* used to fit the time series $u(i)$. Similarly, the minimum mean-squared error $\mathscr{E}_{\min}$, except for a scaling factor, represents an estimate of the white-noise variance σ^2 in the AR model. We may thus use Eq. (6.101) to formulate an *estimate of the AR spectrum* as follows:

$$\hat{S}_{\mathrm{AR}}(\omega) = \frac{\mathscr{E}_{\min}}{\left|1 + \sum_{k=1}^{M} \hat{a}_k^* e^{-j\omega k}\right|^2} \tag{11.88}$$

where the $\hat{a}_k$ are the elements of the vector $\hat{\mathbf{a}}$; the leading element $\hat{a}_0$ of the vector $\hat{\mathbf{a}}$ is equal to unity, by definition. We may also express $\hat{S}_{\mathrm{AR}}(\omega)$ as

$$\hat{S}_{\mathrm{AR}}(\omega) = \frac{\mathscr{E}_{\min}}{|\hat{\mathbf{a}}^H \mathbf{s}(\omega)|^2} \tag{11.89}$$

[6]The correlation matrix $\mathbf{\Phi}$ of Eq. (11.83) does *not* possess a Toeplitz structure. Accordingly, we cannot use the Levinson recursion to develop a fast solution of the augmented normal equations (11.85), as was the case with the augmented Wiener–Hopf equations for stationary inputs. However, Marple (1980, 1981) describes fast recursive algorithms for the efficient solution of the augmented normal equations (11.85). Marple exploits the special Toeplitz structure of the correlation matrix $\mathbf{\Phi}$. The computational complexity of Marple's fast algorithm is proportional to M^2. When the predictor order M is large, the use of Marple's algorithm results in significant savings in computation.

where $\mathbf{s}(\omega)$ is a *variable-frequency vector* or *frequency scanning vector*:

$$\mathbf{s}(\omega) = [1, e^{-j\omega}, \ldots, e^{-j\omega M}]^T, \qquad -\pi < \omega \leq \pi \tag{11.90}$$

Intuitively, the model order M should be as large as possible in order to have a large aperture for the predictor. However, in applying the FBLP algorithm the use of large values of M gives rise to spurious spectral peaks in the AR spectrum. For best performance of the FBLP algorithm, Lang and McClellan (1980) suggest the value

$$M \approx \frac{N}{3} \tag{11.91}$$

where N is the data length.

MVDR Spectrum Estimation

In the method of least squares, as described up to this point in our discussion, there are no *constraints* imposed on the solution. In certain applications, however, the use of such an approach may be unsatisfactory, in which case we may resort to a *constrained* version of the method of least squares. For example, in *adaptive beamforming* that involves spatial processing, we may wish to *minimize the variance (i.e., average power) of the beamformer output while a distortionless response is maintained along the direction of a target signal of interest*. Correspondingly, in the temporal counterpart to this problem, we may be required to *minimize the average power of the spectrum estimator, while a distortionless response is maintained at a particular frequency*. In such applications, the resulting solution is referred to as a *minimum-variance distortionless response (MVDR) estimator* for obvious reasons. To be consistent with the material presented heretofore, we will formulate the temporal version of the MVDR algorithm.

Consider then a linear transversal filter, as depicted in Fig. 11.7. Let the filter output be denoted by $y(i)$. This output is in response to the tap inputs $u(i)$, $u(i-1)$, . . . , $u(i-M)$. Specifically, we have

$$y(i) = \sum_{t=0}^{M} a_t^* u(i-t) \tag{11.92}$$

where $a_0, a_1, \ldots, a_m$ are the transversal filter coefficients. Note, however, that unlike the prediction-error filter of Fig. 11.6, there is no restriction on the filter coefficient a_0; the only reason for using the same terminology as in Fig. 11.6 is because of a desire to be consistent. The requirement is to minimize the *output energy* (assuming the use of the covariance method of data windowing):

$$\mathscr{E}_{\text{out}} = \sum_{i=M+1}^{N} |y(i)|^2$$

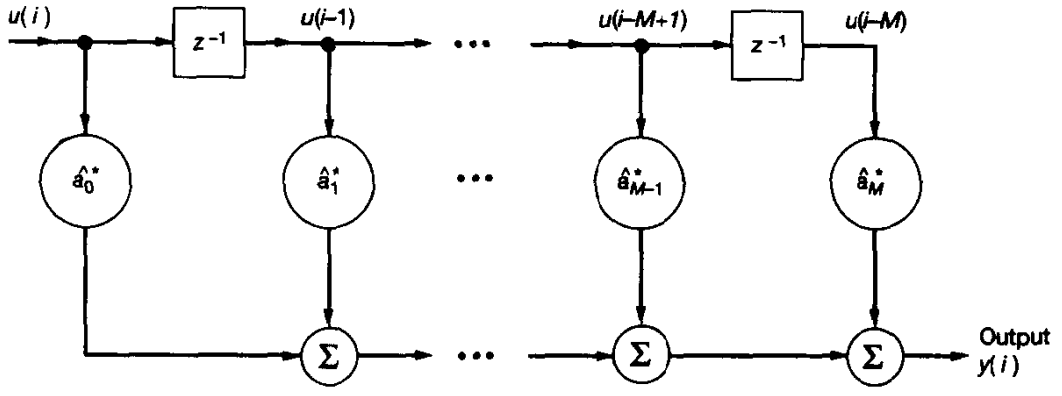

Figure 11.7 Transversal filter.

subject to the *constraint*

$$\sum_{k=0}^{M} a_k^* e^{-jk\omega_0} = 1 \tag{11.93}$$

where ω_0 is an angular frequency of special interest. As in the conventional method of least squares, the filter coefficients $a_0, a_1, \ldots, a_M$ are held constant for the observation interval $1 \leq i \leq N$, where N is the total data length.

To solve this *constrained minimization* problem, we use the *method of Lagrange multipliers*.[7] Specifically, we define the *constrained cost function*

$$\mathscr{C} = \underbrace{\sum_{i=M+1}^{N} |y(i)|^2}_{\text{output energy}} + \underbrace{\lambda\left(\sum_{k=0}^{M} a_k^* e^{-jk\omega_0} - 1\right)}_{\text{linear constraints}} \tag{11.94}$$

where λ is a *complex Lagrange multiplier*. Note that in the constrained approach described herein, there is *no* desired response; in place of it, however, we have a set of linear constraints. Note also that in the absence of a desired response and therefore no frame of reference, the principle of orthogonality loses its meaning in this new setting.

To solve for the optimum values of the filter coefficients, we first determine the gradient vector $\nabla\mathscr{C}$ and then set it equal to zero. Thus, proceeding in a manner similar to that described in Section 11.3, we find that the kth element of the gradient vector for the constrained cost function of Eq. (11.94) is

$$\nabla_k\mathscr{C} = 2\sum_{i=M+1}^{N} u(i-k)y^*(i) + \lambda^* e^{-jk\omega_0} \tag{11.95}$$

[7]The method of Lagrange multipliers is described in Appendix C.

Next, substituting Eq. (11.92) in (11.95), and rearranging terms, we get

$$\nabla_k \mathscr{E} = 2 \sum_{t=0}^{M} a_t \sum_{i=M+1}^{N} u(i-k)u^*(i-t) + \lambda^* e^{-jk\omega_0}$$

$$= 2 \sum_{t=0}^{M} a_t \phi(t, k) + \lambda^* e^{-jk\omega_0} \tag{11.96}$$

where, in the first term of the second line, we have made use of the definition of Eq. (11.29) for the time-averaged autocorrelation function $\phi(t, k)$ of the tap inputs. To minimize the constrained cost function $\mathscr{E}$, we set

$$\nabla_k \mathscr{E} = 0, \qquad k = 0, 1, \ldots, M \tag{11.97}$$

Accordingly, we find from Eq. (11.96) that the tap-weights of the *optimized* transversal filter satisfy the following system of $M + 1$ simultaneous equations:

$$\sum_{t=0}^{M} \hat{a}_t \phi(t, k) = -\frac{1}{2} \lambda^* e^{-jk\omega_0}, \qquad k = 0, 1, \ldots, M \tag{11.98}$$

Using matrix notation, we may rewrite this system of equations in the compact form

$$\mathbf{\Phi}\hat{\mathbf{a}} = -\frac{1}{2} \lambda^* \mathbf{s}(\omega_0) \tag{11.99}$$

where $\mathbf{\Phi}$ is the $(M + 1)$-by-$(M + 1)$ time-averaged correlation matrix of the tap inputs; $\hat{\mathbf{a}}$ is the $(M\ 1)$-by-1 vector of optimum tap weights; and $\mathbf{s}(\omega_0)$ is the $(M + 1)$-by-1 *fixed frequency vector*:

$$\mathbf{s}(\omega_0) = [1, e^{-j\omega_0}, \ldots, e^{-jM\omega_0}]^T \tag{11.100}$$

Assuming $\mathbf{\Phi}$ is nonsingular and therefore its inverse $\mathbf{\Phi}^{-1}$ exists, we may solve Eq. (11.99) for the optimum tap-weight vector:

$$\hat{\mathbf{a}} = -\frac{1}{2} \lambda^* \mathbf{\Phi}^{-1} \mathbf{s}(\omega_0) \tag{11.101}$$

There only remains the problem of evaluating the Lagrange multiplier λ. To solve for λ, we use the linear constraint in Eq. (11.93) for the optimized transversal filter, written in matrix form as

$$\hat{\mathbf{a}}^H \mathbf{s}(\omega_0) = 1 \tag{11.102}$$

Hence, evaluating the *inner product* of the vector $\mathbf{s}_0$ and the vector $\hat{\mathbf{a}}$ in Eq. (11.101), setting the result equal to 1 and solving for λ, we get

$$\lambda^* = -\frac{2}{\mathbf{s}^H(\omega_0)\mathbf{\Phi}^{-1}\mathbf{s}(\omega_0)} \tag{11.103}$$

Finally, substituting this value of λ in Eq. (11.101), we get the MVDR solution:[8]

$$\hat{\mathbf{a}} = \frac{\boldsymbol{\Phi}^{-1}\mathbf{s}(\omega_0)}{\mathbf{s}^H(\omega_0)\boldsymbol{\Phi}^{-1}\mathbf{s}(\omega_0)} \tag{11.104}$$

Thus, given the time-averaged correlation matrix $\boldsymbol{\Phi}$ of the tap inputs and the frequency vector $\mathbf{s}(\omega_0)$, we may use the *MVDR formula* of (11.104) to compute the optimum tap-weight vector $\hat{\mathbf{a}}$ of the transversal filter in Fig. 11.7.

Let $\hat{S}_{\text{MVDR}}(\omega_0)$ denote the minimum value of the output energy $\mathscr{E}_{\text{out}}$, which results when the MVDR solution $\hat{\mathbf{a}}$ of Eq. (11.104) is used for the tap-weight vector under the condition that the response is tuned to the angular frequency ω_0. We may then write

$$\hat{S}_{\text{MVDR}}(\omega_0) = \hat{\mathbf{a}}^H\boldsymbol{\Phi}\hat{\mathbf{a}} \tag{11.105}$$

Substituting Eq. (11.104) in (11.105), and then simplifying the result, we finally get

$$\hat{S}_{\text{MVDR}}(\omega_0) = \frac{1}{\mathbf{s}^H(\omega_0)\boldsymbol{\Phi}^{-1}\mathbf{s}(\omega_0)} \tag{11.106}$$

Equation (11.106) may be given a more general interpretation. Suppose that we define a frequency-scanning vector $\mathbf{s}(\omega)$ as in Eq. (11.90), where the angular frequency ω is now variable in the interval $(-\pi, \pi]$. For each ω, let the tap-weight vector of the transversal filter be assigned a corresponding MVDR estimate. The output energy of the optimized filter then becomes a function of ω. Let $\hat{S}_{\text{MVDR}}(\omega)$ describe this functional dependence, and so we may write[9]

$$\hat{S}_{\text{MVDR}}(\omega) = \frac{1}{\mathbf{s}^H(\omega)\boldsymbol{\Phi}^{-1}\mathbf{s}(\omega)} \tag{11.107}$$

We refer to Eq. (11.107) as the *MVDR spectrum estimate*, and the solution given in Eq. (11.104) as the *MVDR estimate* of the tap-weight vector. Note that at any ω, power due to other frequencies is minimized. Hence, the MVDR spectrum computed in accordance with Eq. (11.107) exhibits relatively sharp peaks.

The MVDR spectrum and AR spectrum are commonly referred to as *super-resolution* or *high-resolution spectra*, in the sense that they both exhibit sub-Rayleigh resolution as power spectrum estimators. For the numerical computation of these spectra, and linear least-squares solutions in general, the recommended procedure is to use a technique known as singular value decomposition, which is considered next.

[8]Equation (11.104) is of the same form as that of Eq. (5.97), except for the use of the time-averaged correlation matrix $\boldsymbol{\Phi}$ in place of the ensemble-averaged correlation matrix $\mathbf{R}$, and the use of symbol $\mathbf{a}$ in place of $\mathbf{w}_o$ for the tap-weight vector.

[9]The method for computing the spectrum in Eq. (11.107) is also referred to in the literature as *Capon's method* (Capon, 1969). The term "minimum-variance distortionless response" owes its origin to Owsley (1984).

11.10 SINGULAR VALUE DECOMPOSITION

The analytic power of *singular-value decomposition* lies in the fact that it applies to square as well as rectangular matrices, be they real or complex. As such, it is extremely well suited for the numerical solution of linear least-squares problems in the sense that *it can be applied directly to the data matrix.*

In Sections 11.5 and 11.7 we described two different forms of the normal equations for computing the linear least-squares solution:

1. The form given in Eq. (11.36), namely,

$$\hat{\mathbf{w}} = \boldsymbol{\Phi}^{-1}\mathbf{z}$$

where $\hat{\mathbf{w}}$ is the least-squares estimate of the tap-weight vector of a transversal filter model, $\boldsymbol{\Phi}$ is the time-averaged correlation matrix of the tap inputs, and $\mathbf{z}$ is the time-averaged cross-correlation vector between the tap inputs and some desired response.

2. The form given in Eq. (11.48) *directly in terms of data matrices*, namely,

$$\hat{\mathbf{w}} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{d}$$

where $\mathbf{A}$ is the data matrix representing the time evolution of the tap input vectors, and $\mathbf{d}$ is the desired data vector representing the time evolution of the desired response.

These two forms are indeed mathematically equivalent. Yet they point to different computational procedures for evaluating the least-squares solution $\hat{\mathbf{w}}$. Equation (11.36) requires knowledge of the time-averaged correlation matrix $\boldsymbol{\Phi}$ that involves computing the product of $\mathbf{A}^H$ and $\mathbf{A}$. On the other hand, in Eq. (11.48) the entire term $(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}$ can be interpreted, in terms of the singular-value decomposition applied directly to the data matrix $\mathbf{A}$, in such a way that the solution computed for $\hat{\mathbf{w}}$ has *twice the number of correct digits* as the solution computed by means of Eq. (11.36) for the same numerical precision. To be specific, define the matrix

$$\mathbf{A}^+ = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H \tag{11.108}$$

Then we may rewrite Eq. (11.36) simply as

$$\hat{\mathbf{w}} = \mathbf{A}^+\mathbf{d} \tag{11.109}$$

The matrix $\mathbf{A}^+$ is called the *pseudoinverse* or the *Moore-Penrose generalized inverse* of the matrix $\mathbf{A}$ (Stewart, 1973; Golub and Van Loan 1989). Equation (11.109) represents a convenient way of saying that "the vector $\hat{\mathbf{w}}$ solves the linear least-squares problem." Indeed, it was with the simple format of Eq. (11.109) in mind and also the desire to be consistent with definitions of the time-averaged correlation matrix $\boldsymbol{\Phi}$ and the cross-correlation vector $\mathbf{z}$ used in Section 11.5 that we defined the data matrix $\mathbf{A}$ and the desired data vector $\mathbf{d}$ in the manner shown in Eqs. (11.44) and (11.46), respectively.

In practice, we often find that the data matrix $\mathbf{A}$ contains linearly dependent columns. Consequently, we are faced with a new situation where we now have to decide on which of an infinite number of possible solutions to the least-squares problem to work with. This issue can indeed be resolved by using the singular-value decomposition technique as described in Section 11.12, even when null($\mathbf{A}$) $\neq \varnothing$, where $\varnothing$ denotes the *null set*.

The Singular-Value Decomposition Theorem

The *singular-value decomposition (SVD)* of a matrix is one of the most elegant algorithms in numerical algebra for providing quantitative information about the structure of a system of linear equations (Klema and Laub, 1980). The system of linear equations that is of specific interest to us is described by

$$\mathbf{A}\hat{\mathbf{w}} = \mathbf{d} \tag{11.110}$$

in which $\mathbf{A}$ is a K-by-M matrix, $\mathbf{d}$ is a K-by-1 vector, and $\hat{\mathbf{w}}$ (representing an estimate of the unknown parameter vector) is an M-by-1 vector. Equation (11.110) represents a simplified matrix form of the normal equations. In particular, premultiplication of both sides of the equation by the vector $\mathbf{A}^H$ yields the normal equations for the least-squares weight vector $\hat{\mathbf{w}}$.

Given the data matrix $\mathbf{A}$, there are two unitary matrices $\mathbf{V}$ and $\mathbf{U}$, such that we may write

$$\mathbf{U}^H\mathbf{A}\mathbf{V} = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{11.111}$$

where $\mathbf{\Sigma}$ is a diagonal matrix:

$$\mathbf{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_W) \tag{11.112}$$

The σ's are ordered as $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_W > 0$. Equation (11.111) is a mathematical statement of the *singular-value decomposition theorem*. This theorem is also referred to as the *Autonne–Eckart–Young theorem* in recognition of its originators.[10]

Figure 11.8 presents a diagrammatic interpretation of the singular value decomposition theorem, as described in Eq. (11.111). In this diagram we have assumed that the number of rows K contained in the data matrix $\mathbf{A}$ is larger than the number of columns M, and that the number of nonzero singular values W is less than M. We may of course restructure the diagrammatic interpretation of the singular value decomposition theorem by expressing the data matrix in terms of the unitary matrices $\mathbf{U}$ and $\mathbf{V}$, and the diagonal matrix $\mathbf{\Sigma}$; this is left as an exercise for the reader.

[10]According to DeMoor and Golub (1989), the singular-value decomposition was introduced in its general form by Autonne in 1902, and an important characterization of it was described by Eckart and Young (1936). For additional notes on the history of the singular-value decomposition, see Klema and Laub (1980).

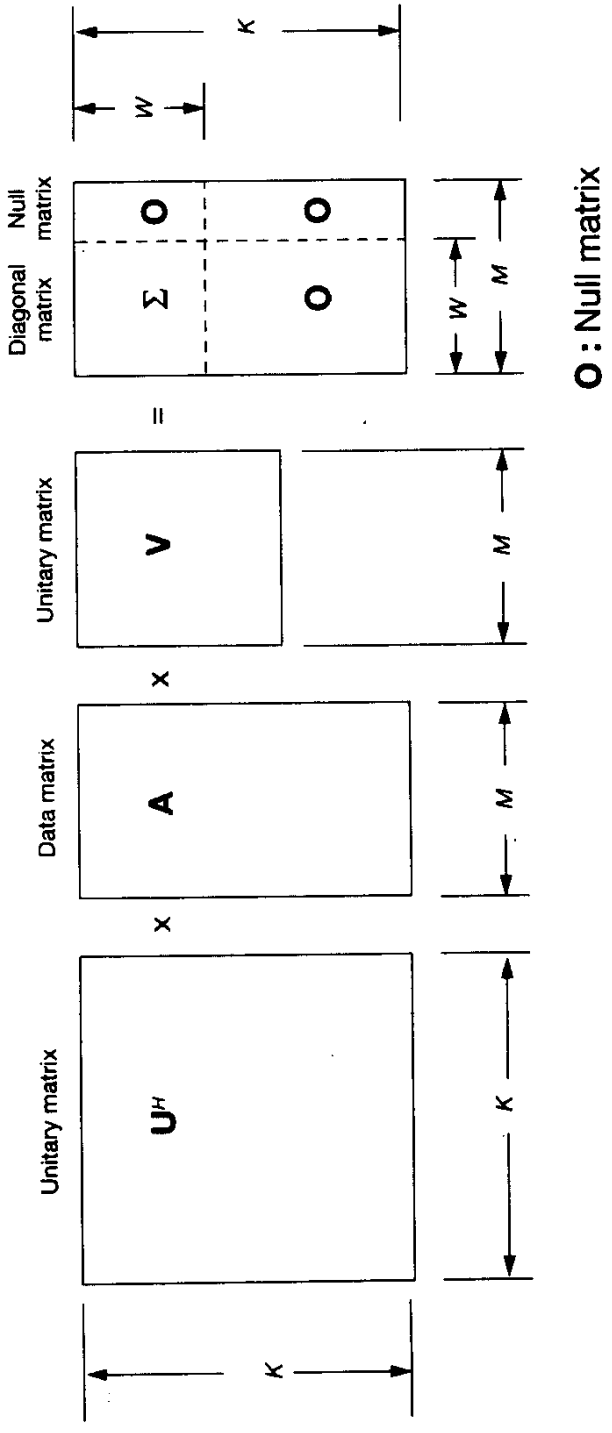

Figure 11.8 Diagrammatic interpretation of the singular value decomposition theorem.

The subscript W in Eq. (11.112) is the *rank* of matrix $\mathbf{A}$, written as rank($\mathbf{A}$); it is defined as the number of linearly independent columns in the matrix $\mathbf{A}$. Note that we always have rank($\mathbf{A}^H$) = rank($\mathbf{A}$).

Since it is possible to have $K > M$ or $K < M$, there are two distinct cases to be considered. We prove the singular-value decomposition theorem by considering both cases, independently of each other. For the case when $K > M$, we have an *overdetermined system* in that we have more equations than unknowns. On the other hand, when $K < M$, we have an *underdetermined system* in that we have more unknowns than equations. In the sequel, we consider these two cases in turn.

Case 1: Overdetermined System. For the case when $K > M$, we form the M-by-M matrix $\mathbf{A}^H\mathbf{A}$ by premultiplying the matrix $\mathbf{A}$ by its Hermitian transpose $\mathbf{A}^H$. Since the matrix $\mathbf{A}^H\mathbf{A}$ is Hermitian and nonnegative definite, its eigenvalues are all real nonnegative numbers. Let these eigenvalues be denoted by $\sigma_1^2, \sigma_2^2, \ldots, \sigma_M^2$, where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_W > 0$, and $\sigma_{W+1}, \sigma_{W+2}, \ldots$ are all zero, where $1 \leq W \leq M$. The matrix $\mathbf{A}^H\mathbf{A}$ has the same rank as $\mathbf{A}$; hence, there are W nonzero eigenvalues. Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M$ denote a set of orthonormal eigenvectors of $\mathbf{A}^H\mathbf{A}$ that are associated with the eigenvalues $\sigma_1^2, \sigma_2^2, \ldots, \sigma_M^2$, respectively. Also, let $\mathbf{V}$ denote the M-by-M unitary matrix whose columns are made up of the eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M$. Thus, using the eigendecomposition of the matrix $\mathbf{A}^H\mathbf{A}$, we may write

$$\mathbf{V}^H\mathbf{A}^H\mathbf{A}\mathbf{V} = \begin{bmatrix} \mathbf{\Sigma}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{11.113}$$

Let the unitary matrix $\mathbf{V}$ be partitioned as

$$\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2] \tag{11.114}$$

where $\mathbf{V}_1$ is an M-by-W matrix,

$$\mathbf{V}_1 = [\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_W] \tag{11.115}$$

and $\mathbf{V}_2$ is an M-by-$(M - W)$ matrix,

$$\mathbf{V}_2 = [\mathbf{v}_{W+1}, \mathbf{v}_{W+2}, \ldots, \mathbf{v}_M] \tag{11.116}$$

with

$$\mathbf{V}_1^H\mathbf{V}_2 = \mathbf{0} \tag{11.117}$$

We may therefore make two deductions from Eq. (11.113):

1. For matrix $\mathbf{V}_1$, we have

$$\mathbf{V}_1^H\mathbf{A}^H\mathbf{A}\mathbf{V}_1 = \mathbf{\Sigma}^2$$

Consequently,

$$\mathbf{\Sigma}^{-1}\mathbf{V}_1^H\mathbf{A}^H\mathbf{A}\mathbf{V}_1\mathbf{\Sigma}^{-1} = \mathbf{I} \tag{11.118}$$

2. For matrix $\mathbf{V}_2$, we have

$$\mathbf{V}_2^H \mathbf{A}^H \mathbf{A} \mathbf{V}_2 = \mathbf{0}$$

Consequently,

$$\mathbf{A}\mathbf{V}_2 = \mathbf{0} \tag{11.119}$$

We now define a new K-by-W matrix

$$\mathbf{U}_1 = \mathbf{A}\mathbf{V}_1\boldsymbol{\Sigma}^{-1} \tag{11.120}$$

Then, from Eq. (11.118) it follows that

$$\mathbf{U}_1^H \mathbf{U}_1 = \mathbf{I} \tag{11.121}$$

which means that the columns of the matrix $\mathbf{U}_1$ are orthonormal with respect to each other. Next, we choose another K-by-$(K - W)$ matrix $\mathbf{U}_2$ such that the K-by-K matrix formed from $\mathbf{U}_1$ and $\mathbf{U}_2$, namely,

$$\mathbf{U} = [\mathbf{U}_1, \mathbf{U}_2] \tag{11.122}$$

is a unitary matrix. This means that

$$\mathbf{U}_1^H \mathbf{U}_2 = \mathbf{0} \tag{11.123}$$

Accordingly, we may use Eqs. (11.114), (11.122), (11.119), (11.120), and (11.123), in that order, and so write

$$\begin{aligned}
\mathbf{U}^H \mathbf{A} \mathbf{V} &= \begin{bmatrix} \mathbf{U}_1^H \\ \mathbf{U}_2^H \end{bmatrix} \mathbf{A}[\mathbf{V}_1, \mathbf{V}_2] \\
&= \begin{bmatrix} \mathbf{U}_1^H \mathbf{A}\mathbf{V}_1 & \mathbf{U}_1^H \mathbf{A}\mathbf{V}_2 \\ \mathbf{U}_2^H \mathbf{A}\mathbf{V}_1 & \mathbf{U}_2^H \mathbf{A}\mathbf{V}_2 \end{bmatrix} \\
&= \begin{bmatrix} (\boldsymbol{\Sigma}^{-1}\mathbf{V}_1^H \mathbf{A}^H)\mathbf{A}\mathbf{V}_1 & \mathbf{U}_1^H(\mathbf{0}) \\ \mathbf{U}_2^H(\mathbf{U}_1\boldsymbol{\Sigma}) & \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}
\end{aligned}$$

which proves Eq. (11.111) for the overdetermined case.

Case 2: Underdetermined System. Consider next the case when $K < M$. This time we form the K-by-K matrix $\mathbf{A}\mathbf{A}^H$ by postmultiplying the matrix $\mathbf{A}$ by its Hermitian transpose $\mathbf{A}^H$. The matrix $\mathbf{A}\mathbf{A}^H$ is also Hermitian and nonnegative definite, so its eigenvalues are likewise real nonnegative numbers. The nonzero eigenvalues of $\mathbf{A}\mathbf{A}^H$ are the *same* as those of $\mathbf{A}^H\mathbf{A}$. We may therefore denote the eigenvalues of $\mathbf{A}\mathbf{A}^H$ as σ_1^2, $\sigma_2^2, \ldots, \sigma_K^2$, where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_W > 0$, and $\sigma_{W+1}, \sigma_{W+2}, \ldots$ are all zero, where

$1 \leq W \leq K$. Let $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_K$ denote a set of orthonormal eigenvctors of the matrix $\mathbf{A}\mathbf{A}^H$ that are associated with the eigenvalues $\sigma_1^2, \sigma_2^2, \ldots, \sigma_K^2$, respectively. Also, let $\mathbf{U}$ denote the unitary matrix whose columns are made up of the eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_K$. Thus, using the eigendecomposition of $\mathbf{A}\mathbf{A}^H$, we may write

$$\mathbf{U}^H\mathbf{A}\mathbf{A}^H\mathbf{U} = \begin{bmatrix} \mathbf{\Sigma}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{11.124}$$

Let the unitary matrix $\mathbf{U}$ be partitioned as

$$\mathbf{U} = [\mathbf{U}_1, \mathbf{U}_2] \tag{11.125}$$

where

$$\mathbf{U}_1 = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_W] \tag{11.126}$$

$$\mathbf{U}_2 = [\mathbf{u}_{W+1}, \mathbf{u}_{W+2}, \ldots, \mathbf{u}_K] \tag{11.127}$$

and

$$\mathbf{U}_1^H\mathbf{U}_2 = \mathbf{0} \tag{11.128}$$

We may therefore make two deductions from Eq. (11.124):

1. For matrix $\mathbf{U}_1$, we have

$$\mathbf{U}_1^H\mathbf{A}\mathbf{A}^H\mathbf{U}_1 = \mathbf{\Sigma}^2$$

Consequently,

$$\mathbf{\Sigma}^{-1}\mathbf{U}_1^H\mathbf{A}\mathbf{A}^H\mathbf{U}_1\mathbf{\Sigma}^{-1} = \mathbf{I} \tag{11.129}$$

2. For matrix $\mathbf{U}_2$, we have

$$\mathbf{U}_2^H\mathbf{A}\mathbf{A}^H\mathbf{U}_2 = \mathbf{0}$$

Consequently,

$$\mathbf{A}^H\mathbf{U}_2 = \mathbf{0} \tag{11.130}$$

We now define an M-by-W matrix

$$\mathbf{V}_1 = \mathbf{A}^H\mathbf{U}_1\mathbf{\Sigma}^{-1} \tag{11.131}$$

Then from Eq. (11.129), it follows that

$$\mathbf{V}_1^H\mathbf{V}_1 = \mathbf{I} \tag{11.132}$$

which means that the columns of the matrix $\mathbf{V}_1$ are orthonormal with respect to each other. Next, we choose another M-by-$(M - W)$ matrix $\mathbf{V}_2$ such that the M-by-M matrix formed from $\mathbf{V}_1$ and $\mathbf{V}_2$, namely,

$$\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2] \tag{11.133}$$

is a unitary matrix. This means that

$$\mathbf{V}_2^H\mathbf{V}_1 = \mathbf{0} \tag{11.134}$$

Accordingly, we may use Eqs. (11.125), (11.133), (11.130), (11.131), and (11.134), in that order, and so write

$$\begin{aligned}\mathbf{U}^H\mathbf{A}\mathbf{V} &= \begin{bmatrix}\mathbf{U}_1^H\\ \mathbf{U}_2^H\end{bmatrix}\mathbf{A}[\mathbf{V}_1, \mathbf{V}_2]\\ &= \begin{bmatrix}\mathbf{U}_1^H\mathbf{A}\mathbf{V}_1 & \mathbf{U}_1^H\mathbf{A}\mathbf{V}_2\\ \mathbf{U}_2^H\mathbf{A}\mathbf{V}_1 & \mathbf{U}_2^H\mathbf{A}\mathbf{V}_2\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{U}_1^H\mathbf{A}(\mathbf{A}^H\mathbf{U}_1\boldsymbol{\Sigma}^{-1}) & (\boldsymbol{\Sigma}\mathbf{V}_1^H)\mathbf{V}_2\\ \end{bmatrix}\\ &= \begin{bmatrix}\boldsymbol{\Sigma} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\end{aligned}$$

This proves Eq. (11.111) for the underdetermined case, and with it the proof of the singular-value decomposition (SVD) theorem is completed.

Terminology and Relation to Eigenanalysis

The numbers $\sigma_1, \sigma_2, \ldots, \sigma_W$, constituting the diagonal matrix $\boldsymbol{\Sigma}$, are called the *singular values* of the matrix $\mathbf{A}$. The columns of the unitary matrix $\mathbf{V}$, that is, $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M$, are the *right singular vectors* of $\mathbf{A}$, and the columns of the second unitary matrix $\mathbf{U}$, that is, $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_K$ are the *left singular vectors* of $\mathbf{A}$. We note from the preceding discussion that the right singular vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M$ are eigenvectors of $\mathbf{A}^H\mathbf{A}$, whereas the left singular vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_K$ are eigenvectors of $\mathbf{A}\mathbf{A}^H$. Note that the number of positive singular values is equal to the rank of the data matrix $\mathbf{A}$. The singular-value decomposition therefore provides the basis of a practical method for determining the rank of a matrix.

Since $\mathbf{U}\mathbf{U}^H$ equals the identity matrix, we find from Eq. (11.111) that

$$\mathbf{A}\mathbf{V} = \mathbf{U}\begin{bmatrix}\boldsymbol{\Sigma} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}$$

It follows therefore that

$$\begin{aligned}\mathbf{A}\mathbf{v}_i &= \sigma_i\mathbf{u}_i, \qquad i = 1, 2, \ldots, W\\ \mathbf{A}\mathbf{v}_i &= \mathbf{0}, \qquad i = W+1, \ldots, K\end{aligned} \tag{11.135}$$

Correspondingly, we may express the data matrix $\mathbf{A}$ in the expanded form

$$\mathbf{A} = \sum_{i=1}^{W}\sigma_i\mathbf{u}_i\mathbf{v}_i^H \tag{11.136}$$

Since $\mathbf{V}\mathbf{V}^H$ equals the identity matrix, we also find from Eq. (11.111) that

$$\mathbf{U}^H\mathbf{A} = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{V}^H$$

or, equivalently,

$$\mathbf{A}^H\mathbf{U} = \mathbf{V}\begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

It follows therefore that

$$\begin{aligned} \mathbf{A}^H\mathbf{u}_i &= \sigma_i\mathbf{v}_i, \qquad i = 1, 2, \ldots, W \\ \mathbf{A}^H\mathbf{u}_i &= \mathbf{0}, \qquad i = W + 1, \ldots, M \end{aligned} \tag{11.137}$$

In this case, we may express the Hermitian transpose of the data matrix $\mathbf{A}$ in the expanded form

$$\mathbf{A}^H = \sum_{i=1}^{W} \sigma_i\mathbf{v}_i\mathbf{u}_i^H \tag{11.138}$$

which checks exactly with Eq. (11.136), and so it should.

Example 2

In this example, we use the SVD to deal with the different facets of *matrix rank*. To be specific, let $\mathbf{A}$ be a K-by-M data matrix with rank W. The matrix $\mathbf{A}$ is said to be of *full rank* if

$$W = \min(K, M)$$

Otherwise, the matrix $\mathbf{A}$ is *rank deficient*. As mentioned previously, the rank W is simply the number of nonzero singular values of matrix $\mathbf{A}$.

Consider next a computational environment that yields a numerical value for each element of the matrix $\mathbf{A}$ that is accurate to within $\pm\epsilon$. Let $\mathbf{B}$ denote the approximate value of matrix $\mathbf{A}$ so obtained. We define the ϵ-*rank* of matrix $\mathbf{A}$ as follows (Golub and Van Loan, 1989):

$$\text{rank}(\mathbf{A}, \epsilon) = \min_{\|\mathbf{A}-\mathbf{B}\|<\epsilon} \text{rank}(\mathbf{B}) \tag{11.139}$$

where $\|\mathbf{A} - \mathbf{B}\|$ is the *spectral norm* of the error matrix $\mathbf{A} - \mathbf{B}$ that results from the use of inaccurate computations. Extending the definition of spectral norm of the matrix introduced in Chapter 4 to the situation at hand, the spectral norm $\|\mathbf{A} - \mathbf{B}\|$ equals the largest singular value of the difference $\mathbf{A} - \mathbf{B}$. In any event, the K-by-M matrix $\mathbf{A}$ is said to be *numerically rank deficient* if

$$\text{rank}(\mathbf{A}, \epsilon) < \min(K, M)$$

The SVD provides a sensible method for characterizing the ϵ-rank and the numerical rank deficiency of the matrix, because the singular values resulting from its use indicate how close a given matrix **A** is to another matrix **B** of lower rank in a simple fashion.

11.11 PSEUDOINVERSE

Our interest in the singular-value decomposition is to formulate a general definition of pseudoinverse. Let **A** denote a K-by-M matrix that has the singular-value decomposition described in Eq. (11.111). We define the pseudoinverse of the matrix **A** as (Stewart, 1973; Golub and Van Loan, 1989):

$$\mathbf{A}^+ = \mathbf{V}\begin{bmatrix} \mathbf{\Sigma}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{U}^H \tag{11.140}$$

where

$$\mathbf{\Sigma}^{-1} = \text{diag}(\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_W^{-1})$$

and W is the rank of the data matrix **A**. The pseudoinverse $\mathbf{A}^+$ may be expressed in the expanded form:

$$\mathbf{A}^+ = \sum_{i=1}^{W} \frac{1}{\sigma_i}\mathbf{v}_i\mathbf{u}_i^H \tag{11.141}$$

We may identify two special cases that can arise as described next.

Case 1: Overdetermined System. In this case, we have $K > M$, and we assume that the rank W equals M so that the inverse matrix $(\mathbf{A}^H\mathbf{A})^{-1}$ exists. The pseudoinverse of the data matrix **A** is defined by

$$\mathbf{A}^+ = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H \tag{11.142}$$

To show the validity of this special formula, we note from Eqs. (11.118) and (11.120) that

$$(\mathbf{A}^H\mathbf{A})^{-1} = \mathbf{V}_1\mathbf{\Sigma}^{-2}\mathbf{V}_1^H$$

and

$$\mathbf{A}^H = \mathbf{V}_1\mathbf{\Sigma}\mathbf{U}_1^H$$

Therefore, using this pair of relations, we may express the right-hand side of Eq. (11.142) as follows:

$$\begin{aligned}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H &= (\mathbf{V}_1\mathbf{\Sigma}^{-2}\mathbf{V}_1^H)(\mathbf{V}_1\mathbf{\Sigma}\mathbf{U}_1^H) \\ &= \mathbf{V}_1\mathbf{\Sigma}^{-1}\mathbf{U}_1^H \\ &= \mathbf{V}\begin{bmatrix} \mathbf{\Sigma}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{U}^H \\ &= \mathbf{A}^+\end{aligned}$$

Case 2: Underdetermined System. In this second case, we have $M > K$, and we assume that the rank W equals K so that the inverse matrix $(\mathbf{A}\mathbf{A}^H)^{-1}$ exists. The pseudoinverse of the data matrix $\mathbf{A}$ is now defined by

$$\mathbf{A}^+ = \mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1} \tag{11.143}$$

To show the validity of this second special formula, we note from Eqs. (11.129) and (11.131) that

$$(\mathbf{A}\mathbf{A}^H)^{-1} = \mathbf{U}_1\mathbf{\Sigma}^{-2}\mathbf{U}_1^H$$

and

$$\mathbf{A}^H = \mathbf{V}_1\mathbf{\Sigma}\mathbf{U}_1^H$$

Therefore, using this pair of relations in the right-hand side of Eq. (11.143), we get

$$\begin{aligned}\mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1} &= (\mathbf{V}_1\mathbf{\Sigma}\mathbf{U}_1^H)(\mathbf{U}_1\mathbf{\Sigma}^{-2}\mathbf{U}_1^H)\\ &= \mathbf{V}_1\mathbf{\Sigma}^{-1}\mathbf{U}_1^H\\ &= \mathbf{V}\begin{bmatrix}\mathbf{\Sigma}^{-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{U}^H\\ &= \mathbf{A}^+\end{aligned}$$

Note, however, the pseudoinverse $\mathbf{A}^+$ as described in Eq. (11.140) or equivalently, Eq. (11.141) is of general application, in that it applies whether the data matrix $\mathbf{A}$ refers to an overdetermined or an underdetermined system and regardless of what the rank W is. Most importantly, it is numerically stable.

11.12 INTERPRETATION OF SINGULAR VALUES AND SINGULAR VECTORS

Consider a K-by-M data matrix $\mathbf{A}$, for which the singular-value decomposition is given in Eq. (11.111) and the pseudoinverse is correspondingly given in Eq. (11.140). We assume that the system is overdetermined. Define a K-by-1 vector $\mathbf{y}$ and an M-by-1 vector $\mathbf{x}$ that are related to each other by the transformation matrix $\mathbf{A}$, as shown by

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{11.144}$$

The vector $\mathbf{x}$ is constrained to have a Euclidean norm of unity; that is,

$$\|\mathbf{x}\| = 1 \tag{11.145}$$

Given the transformation of Eq. (11.144) and the constraint of Eq. (11.145), we wish to find the resulting locus of the points defined by the vector $\mathbf{y}$ in a K-dimensional space.

Solving Eq. (11.144) for $\mathbf{x}$, we get

$$\mathbf{x} = \mathbf{A}^+\mathbf{y} \tag{11.146}$$

where $\mathbf{A}^+$ is the pseudoinverse of $\mathbf{A}$. Substituting Eq. (11.142) in (11.146), we get

$$\begin{aligned}\mathbf{x} &= \sum_{i=1}^{W} \frac{1}{\sigma_i}\mathbf{v}_i\mathbf{u}_i^H\mathbf{y} \\ &= \sum_{i=1}^{W} \frac{(\mathbf{u}_i^H\mathbf{y})}{\sigma_i}\mathbf{v}_i\end{aligned} \tag{11.147}$$

where W is the rank of matrix $\mathbf{A}$, and the inner product $\mathbf{u}_i^H\mathbf{y}$ is a scalar. Imposing the constraint of Eq. (11.145) on (11.147), and recognizing that the right singular vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_W$ form an orthonormal set, we get

$$\sum_{i=1}^{W} \frac{|\mathbf{y}^H\mathbf{u}_i|^2}{\sigma_i^2} = 1 \tag{11.148}$$

Equation (11.148) defines the locus traced out by the tip of vector $\mathbf{y}$ in a K-dimensional space. Indeed, this is the equation of a *hyperellipsoid* (Golub and Van Loan, 1989).

To see this interpretation in a better way, define the complex scalar

$$\begin{aligned}\zeta_i &= \mathbf{y}^H\mathbf{u}_i \\ &= \sum_{k=1}^{K} y_k^* u_{ik}, \qquad i = 1, \ldots, W\end{aligned} \tag{11.149}$$

In other words, the complex scalar ζ_i is a linear combination of all possible values of the elements of the left singular vector $\mathbf{u}_i$, so ζ_i is referred to as the "span" of $\mathbf{u}_i$. We may thus rewrite Eq. (11.148) as

$$\sum_{i=1}^{W} \frac{|\zeta_i|^2}{\sigma_i^2} = 1 \tag{11.150}$$

This is the equation of a hyperellipsoid with coordinates $|\zeta_1|, \ldots, |\zeta_W|$ and semi-axis whose lengths are the singular values $\sigma_1, \ldots, \sigma_W$, respectively. Figure 11.9 illustrates the locus traced out by Eq. (11.148) for the case of $W = 2$ and $\sigma_1 > \sigma_2$, assuming that the data matrix A is real.

11.13 MINIMUM NORM SOLUTION TO THE LINEAR LEAST-SQUARES PROBLEM

Having equipped ourselves with the general definition of the pseudoinverse of a matrix $\mathbf{A}$ in terms of its singular-value decomposition, we are now ready to tackle the solution to the linear least-squares problem even when null(A) $\neq \varnothing$. In particular, we define the solution to the least-squares problem as in Eq. (11.109), reproduced here for convenience:

$$\hat{\mathbf{w}} = \mathbf{A}^+\mathbf{d} \tag{11.151}$$

The pseudoinverse matrix $\mathbf{A}^+$ is itself defined by Eq. (11.140). We thus find that, out of the many vectors that solve the least-squares problem when null $(\mathbf{A}) \neq \varnothing$, the one defined

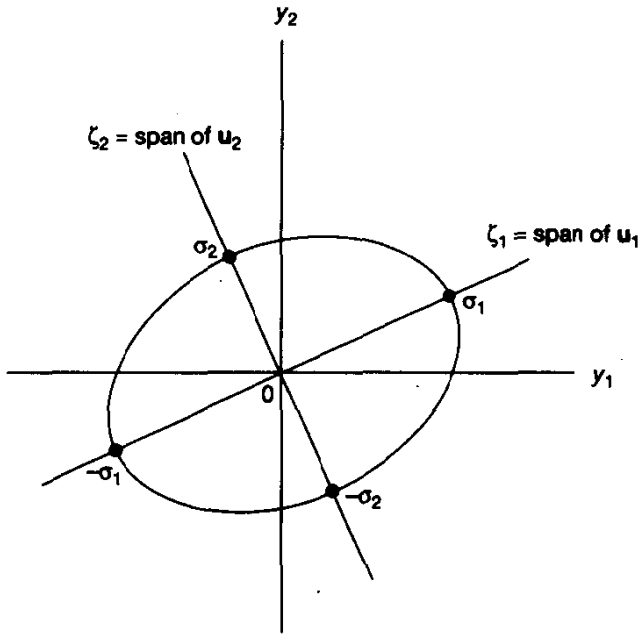

Figure 11.9 Locus of Eq. (11.150) for real data with $W = 2$ and $\sigma_1 > \sigma_2$.

by Eq. (11.151) is *unique* in that it has the *shortest length possible in the Euclidean sense* (Stewart, 1973).

We prove this important result by manipulating the equation that defines the minimum value of the sum of error squares produced in the method of least squares. We note that both matrix products $\mathbf{V}\mathbf{V}^H$ and $\mathbf{U}\mathbf{U}^H$ equal identity matrices. Hence, we may start with Eq. (11.49) and combine it with Eq. (11.48), and then write

$$\begin{aligned}\mathscr{E}_{\min} &= \mathbf{d}^H\mathbf{d} - \mathbf{d}^H\mathbf{A}\hat{\mathbf{w}} \\ &= \mathbf{d}^H(\mathbf{d} - \mathbf{A}\hat{\mathbf{w}}) \\ &= \mathbf{d}^H\mathbf{U}\mathbf{U}^H(\mathbf{d} - \mathbf{A}\mathbf{V}\mathbf{V}^H\hat{\mathbf{w}}) \\ &= \mathbf{d}^H\mathbf{U}(\mathbf{U}^H\mathbf{d} - \mathbf{U}^H\mathbf{A}\mathbf{V}\mathbf{V}^H\hat{\mathbf{w}})\end{aligned} \tag{11.152}$$

Let

$$\begin{aligned}\mathbf{V}^H\hat{\mathbf{w}} &= \mathbf{b} \\ &= \begin{bmatrix}\mathbf{b}_1 \\ \mathbf{b}_2\end{bmatrix}\end{aligned} \tag{11.153}$$

and

$$\begin{aligned}\mathbf{U}^H\mathbf{d} &= \mathbf{c} \\ &= \begin{bmatrix}\mathbf{c}_1 \\ \mathbf{c}_2\end{bmatrix}\end{aligned} \tag{11.154}$$

where $\mathbf{b}_1$ and $\mathbf{c}_1$ are W-by-1 vectors, and $\mathbf{b}_2$ and $\mathbf{c}_2$ are two other vectors. Thus, substituting Eqs. (11.111), (11.153), and (11.154) in (11.152), we get

$$\begin{aligned}\mathscr{E}_{\min} &= \mathbf{d}^H\mathbf{U}\left(\begin{bmatrix}\mathbf{c}_1\\ \mathbf{c}_2\end{bmatrix} - \begin{bmatrix}\boldsymbol{\Sigma} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{bmatrix}\right)\\ &= \mathbf{d}^H\mathbf{U}\begin{bmatrix}\mathbf{c}_1 - \boldsymbol{\Sigma}\mathbf{b}_1\\ \mathbf{c}_2\end{bmatrix}\end{aligned} \tag{11.155}$$

For $\mathscr{E}_{\min}$ to be minimum, we require that

$$\mathbf{c}_1 = \boldsymbol{\Sigma}\mathbf{b}_1 \tag{11.156}$$

or, equivalently,

$$\mathbf{b}_1 = \boldsymbol{\Sigma}^{-1}\mathbf{c}_1 \tag{11.57}$$

We observe that $\mathscr{E}_{\min}$ is independent of $\mathbf{b}_2$. Hence, the value of $\mathbf{b}_2$ is arbitrary. However, if we let $\mathbf{b}_2 = \mathbf{0}$, we get the special result

$$\begin{aligned}\hat{\mathbf{w}} &= \mathbf{V}\mathbf{b}\\ &= \mathbf{V}\begin{bmatrix}\boldsymbol{\Sigma}^{-1}\mathbf{c}_1\\ \mathbf{0}\end{bmatrix}\end{aligned} \tag{11.158}$$

We may also express $\hat{\mathbf{w}}$ in the equivalent form:

$$\begin{aligned}\hat{\mathbf{w}} &= \mathbf{V}\begin{bmatrix}\boldsymbol{\Sigma}^{-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{c}_1\\ \mathbf{c}_2\end{bmatrix}\\ &= \mathbf{V}\begin{bmatrix}\boldsymbol{\Sigma}^{-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{U}^H\mathbf{d}\\ &= \mathbf{A}^+\mathbf{d}\end{aligned}$$

This coincides exactly with the value defined by Eq. (11.151), where the pseudoinverse $\mathbf{A}^+$ is defined by Eq. (11.140). In effect, we have shown that this value of $\hat{\mathbf{w}}$ does indeed solve the linear least-squares problem.

Moreover, the vector $\hat{\mathbf{w}}$ so defined is *unique*, in that it has the minimum Euclidean norm possible. In particular, since $\mathbf{V}\mathbf{V}^H = \mathbf{I}$, we find from Eq. (11.158) that the squared Euclidean norm of $\hat{\mathbf{w}}$ equals

$$\|\hat{\mathbf{w}}\|^2 = \|\boldsymbol{\Sigma}^{-1}\mathbf{c}_1\|^2$$

Consider now another possible solution to the linear least-squares problem that is defined by

$$\mathbf{w}' = \mathbf{V}\begin{bmatrix}\boldsymbol{\Sigma}^{-1}\mathbf{c}_1\\ \mathbf{b}_2\end{bmatrix}, \qquad \mathbf{b}_2 \neq \mathbf{0}$$

The squared Euclidean norm of $\mathbf{w}'$ equals

$$\|\mathbf{w}'\|^2 = \|\boldsymbol{\Sigma}^{-1}\mathbf{c}_1\|^2 + \|\mathbf{b}_2\|^2$$

For any $\mathbf{b}_2 \neq \mathbf{0}$, we see therefore that

$$\|\mathbf{w}\| < \|\mathbf{w}'\| \tag{11.159}$$

In summary, the tap-weight $\hat{\mathbf{w}}$ of a linear transversal filter defined in by Eq. (11.151) is a unique solution to the linear least-squares problem, even when null(A) $\neq \emptyset$. *The vector $\hat{\mathbf{w}}$ is unique in the sense that it is the only tap-weight vector that simultaneously satisfies two requirements:* (1) *it produces the minimum sum of error squares, and* (2) *it has the smallest Euclidean norm possible. This special value of the tap-weight vector $\hat{\mathbf{w}}$ is called the minimum-norm solution.*

Another Formulation of the Minimum-Norm Solution

We may develop an expanded formulation of the minimum-norm solution, depending on whether we are dealing with the overdetermined or underdetermined case. These two cases are considered in turn.

Case 1: Overdetermined. For this case, the number of equations K is greater than the number of unknown parameters M. To proceed then, we substitute Eq. (11.140) in (11.151), and then use the partitioned forms of the unitary matrices $\mathbf{V}$ and $\mathbf{U}$. We may thus write

$$\begin{aligned}\hat{\mathbf{w}} &= (\mathbf{V}_1\boldsymbol{\Sigma}^{-1})(\mathbf{A}\mathbf{V}_1\boldsymbol{\Sigma}^{-1})^H\mathbf{d} \\ &= \mathbf{V}_1\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{V}_1^H\mathbf{A}^H\mathbf{d} \\ &= \mathbf{V}_1\boldsymbol{\Sigma}^{-2}\mathbf{V}_1\mathbf{A}^H\mathbf{d}\end{aligned} \tag{11.160}$$

Hence, using the definition [see Eq. (11.115)]

$$\mathbf{V}_1 = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_W]$$

in Eq. (11.160), we get the following expanded formulation for $\hat{\mathbf{w}}$ for the overdetermined case:

$$\hat{\mathbf{w}} = \sum_{i=1}^{W} \frac{\mathbf{v}_i}{\sigma_i^2}\mathbf{v}_i^H\mathbf{A}^H\mathbf{d} \tag{11.161}$$

Case 2: Underdetermined. For this second case, the number of equations K is smaller than the number of unknowns M. This time we find it appropriate to use the representation given in Eq. (11.131) for the submatrix $\mathbf{V}_1$ in terms of the data matrix $\mathbf{A}$. Thus, substituting Eq. (11.131) in (11.151), we get

$$\begin{aligned}\hat{\mathbf{w}} &= (\mathbf{A}^H\mathbf{U}_1\boldsymbol{\Sigma}^{-1})(\boldsymbol{\Sigma}^{-1}\mathbf{U}_1\mathbf{d}) \\ &= \mathbf{A}^H\mathbf{U}_1\boldsymbol{\Sigma}^{-2}\mathbf{U}_1^H\mathbf{d}\end{aligned} \tag{11.162}$$

Substituting the definition [see Eq. (11.126)]

$$\mathbf{U}_1 = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_W]$$

in Eq. (11.162), we get the following expanded formulation for $\hat{\mathbf{w}}$ for the underdetermined case:

$$\hat{\mathbf{w}} = \sum_{i=1}^{W} \frac{(\mathbf{u}_i^H \mathbf{d})}{\sigma_i^2} \mathbf{A}^H \mathbf{u}_i \tag{11.163}$$

which is different from that of Eq. (11.161) for the overdetermined case.

The important point to note is that the expanded solutions of $\hat{\mathbf{w}}$ given in Eqs. (11.161) and (11.163) for the overdetermined and underdetermined systems, respectively, are both contained in the compact formula of Eq. (11.151). Indeed, from a numerical computation point of view, the use of Eq. (11.151) is the preferred method for computing the least-squares estimatee $\hat{\mathbf{w}}$.

11.14 NORMALIZED LMS ALGORITHM VIEWED AS THE MINIMUM-NORM SOLUTION TO AN UNDERDETERMINED LEAST-SQUARES ESTIMATION PROBLEM

In Chapter 9 we derived the normalized least-mean-square (LMS) algorithm as the solution to a constrained minimization problem. In this section we revisit this algorithm in light of the theory developed on singular-value decomposition. In particular, we show that the normalized LMS algorithm is indeed the minimum-norm solution to an underdetermined linear least-squares problem involving a single error equation with M unknowns, where M is the dimension of the tap-weight vector in the algorithm.

Consider the error equation

$$\epsilon(n) = d(n) - \hat{\mathbf{w}}^H(n+1)\mathbf{u}(n) \tag{11.164}$$

where $d(n)$ is a desired response and $\mathbf{u}(n)$ is a tap-input vector, both measured at time n. The requirement is to find the tap-weight vector $\hat{\mathbf{w}}(n+1)$, measured at time $n+1$, such that the change in the tap-weight vector given by

$$\delta\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n+1) - \hat{\mathbf{w}}(n) \tag{11.165}$$

is minimized, subject to the constraint

$$\epsilon(n) = 0 \tag{11.166}$$

Using Eq. (11.165) in (11.164), we may reformulate the error $\epsilon(n)$ as

$$\epsilon(n) = d(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) - \delta\hat{\mathbf{w}}^H(n+1)\mathbf{u}(n) \tag{11.167}$$

We now recognize the customary definition of the estimation error, namely,

$$e(n) = d(n) - \hat{\mathbf{w}}^H(n)\mathbf{u}(n) \tag{11.168}$$

Hence, we may simplify Eq. (11.168) as

$$\epsilon(n) = e(n) - \delta\hat{\mathbf{w}}^H(n+1)\mathbf{u}(n) \tag{11.169}$$

TABLE 11.1 SUMMARY OF CORRESPONDENCES BETWEEN LINEAR LEAST-SQUARES ESTIMATION AND NORMALIZED LMS ALGORITHM

	Linear least-squares estimation (underdetermined)	Normalized LMS algorithm
Data matrix	$\mathbf{A}$	$\mathbf{u}^H(n)$
Desired data vector	$\mathbf{d}$	$e^*(n)$
Parameter vector	$\hat{\mathbf{w}}$	$\delta\hat{\mathbf{w}}(n+1)$
Rank	W	1
Eigenvalue	$\sigma_i^2, i = 1, \ldots, W$	$\|\mathbf{u}(n)\|^2$
Eigenvector	$\mathbf{u}_i, i = 1, \ldots, W$	1

Thus, complex conjugating both sides of Eq. (11.169), we note that the constraint of Eq. (11.166) is equivalent to

$$\mathbf{u}^H(n)\delta\hat{\mathbf{w}}(n+1) = e^*(n) \tag{11.170}$$

Accordingly, we may restate our constrained minimization problem as follows:

Find the minimum-norm solution for the change $\delta\hat{\mathbf{w}}(n+1)$ in the tap-weight vector at time $n+1$, which satisfies the constraint

$$\mathbf{u}^H(n)\delta\hat{\mathbf{w}}(n+1) = e^*(n)$$

This problem is one of linear least-squares estimation that is underdetermined. To solve it, we may use the method of singular-value decomposition described in Eq. (11.163). To help us in the application of this method, we use Eq. (11.170) to make the identifications listed in Table 11.1 between the normalized LMS algorithm and linear least-squares estimation. In particular, we note that the normalized LMS algorithm has only one nonzero singular value equal to the squared norm of the tap-input vector $\mathbf{u}(n)$; that is, the rank $W = 1$. The corresponding left-singular vector is therefore simply equal to one. Hence, with the aid of Table 11.1, the application of Eq. (11.163) yields

$$\delta\hat{\mathbf{w}}(n+1) = \frac{1}{\|\mathbf{u}(n)\|^2}\,\mathbf{u}(n)e^*(n) \tag{11.171}$$

This is precisely the result that we derived previously in Chapter 9; see Eq. (9.139).

We may next follow a reasoning similar to that described in Section 9.10 and redefine the change $\delta\hat{\mathbf{w}}(n+1)$ by introducing a scaling factor $\tilde{\mu}$ as shown by [see Eq. (9.140)]

$$\delta\hat{\mathbf{w}}(n+1) = \frac{\tilde{\mu}}{\|\mathbf{u}(n)\|^2}\,\mathbf{u}(n)e^*(n)$$

or, equivalently, we may write

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{\tilde{\mu}}{\|\mathbf{u}(n)\|^2}\,\mathbf{u}(n)e^*(n) \tag{11.172}$$

By so doing, we are able to exercise control over the change in the tap-weight vector from one iteration to the next without changing its direction. Equation (11.172) is the tap-weight vector update for the normalized LMS algorithm.

The important point to note from the discussion presented in this section is that the singular-value decomposition provides an insightful link between the underdetermined form of linear least-squares estimation and LMS theory. In particular, we have shown that the weight update in the normalized LMS algorithm may indeed be viewed as the minimum norm solution to an underdetermined form of the linear least-squares problem. The problem involves a single error equation with a number of unknowns equal to the dimension of the tap-weight vector in the algorithm.

11.15 SUMMARY AND DISCUSSION

In this chapter we presented a detailed discussion of the method of least-squares for solving the linear adaptive filtering problem. The distinguishing features of this approach include the following:

- It is a model-dependent procedure that operates on the input data on a block-by-block basis.
- It yields a solution for the tap-weight vector of an adaptive transversal filter that is the best linear unbiased estimate (BLUE), assuming that the measurement error process in the underlying model is white with zero mean.

The method of least squares is well suited for solving super-resolution spectrum estimation/beamforming problems, such as those based on autoregressive (AR) and minimum-variance distortionless response (MVDR) models. For the efficient computation of these spectra, and linear least-squares solution in general, the recommended procedure is to use singular value decomposition (SVD) that operates on the input data directly. The SVD is defined by the following parameters:

- A set of left singular vectors that form a unitary matrix
- A set of right singular vectors that form another unitary matrix
- A corresponding set of nonzero singular values

The important advantage of using the SVD to solve a linear least-squares problem is that the solution, defined in terms of the pseudoinverse of the input data matrix, is numerically stable. An algorithm is said to be *numerically stable* if it does not introduce any more sensitivity to perturbation than that which is inherently present in the problem under study (Klema and Laub, 1980).

Another useful application of the SVD is in *rank determination*. The *column rank* of a matrix is defined by the number of linearly independent columns of the matrix. Specifi-

cally, we say that an M-by-K matrix, with $M \geq K$, has *full* column rank if and only if it has K independent columns. In theory, the issue of full rank determination is a yes–no type of proposition in the sense that either the matrix in question has full rank or it does not. In practice, however, the fuzzy nature of a data matrix and the use of inexact (finite-precision) arithmetic complicate the rank determination problem. The SVD provides a practical method for determining the rank of a matrix, given fuzzy data and roundoff errors due to finite-precision computations.

PROBLEMS

1. Consider a linear array consisting of M uniformly spaced sensors. The output of sensor k observed at time i is denoted by $u(k, i)$ where $k = 1, 2, \ldots, M$ and $i = 1, 2, \ldots, n$. In effect, the observations $u(1, i), u(2, i), \ldots, u(M, i)$ define snapshot i. Let $\mathbf{A}$ denote the n-by-M data matrix, whose Hermitian transpose is defined by

$$\mathbf{A}^H = \begin{bmatrix} u(1,1) & u(1,2) & \cdots & u(1,n) \\ u(2,1) & u(2,2) & \cdots & u(2,n) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ u(M,1) & u(M,2) & \cdots & u(M,n) \end{bmatrix}$$

   where the number of columns equals the number of snapshots, and the number of rows equals the number of sensors in the array. Demonstrate the following interpretations:
   (a) The M-by-M matrix $\mathbf{A}^H\mathbf{A}$ is the *spatial* correlation matrix with temporal averaging. This form of averaging assumes that the environment is temporally stationary.
   (b) The n-by-n matrix $\mathbf{A}\mathbf{A}^H$ is the *temporal* correlation matrix with spatial averaging. This form of averaging assumes that the environment is spatially stationary.

2. We say that the least-squares estimate $\hat{\mathbf{w}}$ is *consistent* if, in the long run, the difference between $\hat{\mathbf{w}}$ and the unknown parameter vector $\mathbf{w}_o$ of the multiple linear regression model becomes negligibly small in the mean-square sense. Hence, show that the least-squares estimate $\hat{\mathbf{w}}$ is consistent if the error vector $\boldsymbol{\epsilon}_o$ has zero mean and its elements are uncorrelated and if the trace of the inverse matrix $\boldsymbol{\Phi}^{-1}$ approaches zero as the number of observations, N, approaches infinity.

3. In Example 1 in Section 11.6, we used a 3-by-2 input data matrix and 3-by-1 desired data vector to illustrate the corollary to the principle of orthogonality. Use the data given in that example to calculate the two tap-weights of the linear least-squares filter.

4. In the autocorrelation method of linear prediction, we choose the tap-weight vector of a transversal predictor to minimize the error energy

$$\mathscr{E}_f = \sum_{n=1}^{\infty} |f(n)|^2$$

   where $f(n)$ is the prediction error. Show that the transfer function $H(z)$ of the (forward) prediction-error filter is minimum phase, in that its roots must lie strictly within the unit circle.

Hints: (1) Express the transfer function $H(z)$ of order M (say) as the product of a simple zero factor $(1 - z_i z^{-1})$ and a function $H'(z)$. Hence, minimize the prediction-error energy with respect to the magnitude of zero z_i.

(2) Use the Cauchy–Schwartz inequality:

$$\mathrm{Re}\left[\sum_{n=1}^{\infty} e^{j\theta} g(n-1)g^*(n)\right] \le \left[\sum_{n=1}^{\infty} |g(n)|^2\right]^{1/2}\left[\sum_{n=1}^{\infty} |e^{j\theta} g(n-1)|^2\right]^{1/2}$$

The equality holds if and only if $g(n) = e^{j\theta} g(n-1)$ for $n = 1, 2, \ldots, \infty$.

5. Figure 11.5(a) shows a *forward linear predictor* using a transversal structure, with the tap inputs $u(i-1), u(i-2), \ldots, u(i-M)$ used to make a linear prediction of $u(i)$. The problem is to find the tap-weight vector $\hat{\mathbf{w}}$ that minimizes the sum of forward prediction-error squares:

$$\mathscr{E}_f = \sum_{i=M+1}^{N} |f_M(i)|^2$$

where $f_M(i)$ is the forward prediction error. Find the following parameters:

(a) The M-by-M correlation matrix of the tap inputs of the predictor.

(b) The M-by-1 cross-correlation vector between the tap inputs of the predictor and the desired response $u(i)$.

(c) The minimum value of $\mathscr{E}_f$.

6. Figure 11.5(b) shows a *backward linear predictor* using a transversal structure, with the tap inputs $u(i-M+1), \ldots, u(i-1), u(i)$ used to make a linear prediction of the input $u(i-M)$. The problem is to find the tap-weight vector $\hat{\mathbf{w}}$ that minimizes the sum of backward prediction-error squares

$$\mathscr{E}_b = \sum_{i=M+1}^{N} |b_M(i)|^2$$

where $b_M(i)$ is the backward prediction error. Find the following parameters:

(a) The M-by-M correlation matrix of the tap inputs.

(b) The M-by-1 correlation vector between the tap inputs and the desired response $u(i-M)$.

(c) The minimum value of $\mathscr{E}_b$.

7. Use a direct approach to derive the system of normal equations given in expanded form in Eq. (11.31).

8. Calculate the singular values and singular vectors of the 2-by-2 real matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & -1 \\ 0.5 & 2 \end{bmatrix}$$

Do the calculation using two different methods:

(a) Eigendecomposition of the matrix product $\mathbf{A}^T\mathbf{A}$.

(b) Eigendecomposition of the matrix product $\mathbf{A}\mathbf{A}^T$.

Hence, find the pseudoinverse of matrix $\mathbf{A}$.

9. Consider the 2-by-2 complex matrix

$$\mathbf{A} = \begin{bmatrix} 1+j & 1+0.5j \\ 0.5-j & 1-j \end{bmatrix}$$

Calculate the singular values and singular vectors of the matrix $\mathbf{A}$ by proceeding as follows:

(a) Construct the matrix $\mathbf{A}^H\mathbf{A}$; hence, evaluate the eigenvalues and eigenvectors of $\mathbf{A}^H\mathbf{A}$.

(b) Construct the matrix $\mathbf{A}\mathbf{A}^H$; hence, evaluate the eigenvalues and eigenvectors of $\mathbf{A}\mathbf{A}^H$.

(c) Relate the eigenvalues and eigenvectors calculated in parts (a) and (b) to the singular values and singular vectors of $\mathbf{A}$.

10. Refer back to Example 1 in Section 11.7. For the sets of data given in that example, do the following:

(a) Calculate the pseudo-inverse of the 3-by-2 data matrix $\mathbf{A}$.

(b) Use this value of the pseudo-inverse $\mathbf{A}^+$ to calculate the two tap weights of the linear least-squares filter.

11. In this problem we explore the derivation of the weight update for the normalized LMS algorithm described in Eq. (9.144) using the idea of singular-value decomposition. This problem may be viewed as an extension of the discussion presented in Section 11.14. Find the minimum norm solution for the coefficient vector

$$\mathbf{c}(n+1) = \begin{bmatrix} \delta\hat{\mathbf{w}}(n+1) \\ 0 \end{bmatrix}$$

that satisfies the equation

$$\mathbf{x}^H(n)\mathbf{c}(n+1) = e^*(n)$$

where

$$\mathbf{x}(n) = \begin{bmatrix} \mathbf{u}(n) \\ \sqrt{a} \end{bmatrix}$$

Hence, show that

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{\tilde{\mu}}{a + \|\mathbf{u}(n)\|^2}\,\mathbf{u}(n)e^*(n)$$

where $a > 0$, and $0 < \tilde{\mu} < 2$. [This is the weight update described in Eq. (9.144).]

12. You are given a processor that is designed to perform the singular-value composition of a K-by-M data matrix $\mathbf{A}$. Using such a processor, develop block diagrams for the following two super-resolution algorithms:

(a) The autoregressive (AR) algorithm

(b) The minimum-variance distortionless response (MVDR) algorithm