Chapter 2
Fundamentals of Adaptive Filtering

2.1 Introduction

This chapter includes a brief review of deterministic and random signal representations. Due to the extent of those subjects, our review is limited to the concepts that are directly relevant to adaptive filtering. The properties of the correlation matrix of the input signal vector are investigated in some detail, since they play a key role in the statistical analysis of the adaptive-filtering algorithms.

The Wiener solution that represents the minimum mean-square error (MSE) solution of discrete-time filters realized through a linear combiner is also introduced. This solution depends on the input signal correlation matrix as well as on the cross-correlation between the elements of the input signal vector and the reference signal. The values of these correlations form the parameters of the MSE surface, which is a quadratic function of the adaptive-filter coefficients. The linearly constrained Wiener filter is also presented, a technique commonly used in antenna array processing applications. The transformation of the constrained minimization problem into an unconstrained one is also discussed. Motivated by the importance of the properties of the MSE surface, we analyze them using some results related to the input signal correlation matrix.

In practice the parameters that determine the MSE surface shape are not available. What is left is to directly or indirectly estimate these parameters using the available data and to develop adaptive algorithms that use these estimates to search the MSE surface, such that the adaptive-filter coefficients converge to the Wiener solution in some sense. The starting point to obtain an estimation procedure is to investigate the convenience of using the classical searching methods of optimization theory [1–3] to adaptive filtering. The Newton and steepest-descent algorithms are investigated as possible searching methods for adaptive filtering. Although both methods are not directly applicable to practical adaptive filtering, smart reflections inspired on them led to practical algorithms such as the least-mean-square (LMS)

P.S.R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*,
DOI 10.1007/978-1-4614-4106-9_2, © Springer Science+Business Media New York 2013

[4, 5] and Newton-based algorithms. The Newton and steepest-descent algorithms are introduced in this chapter, whereas the LMS algorithm is treated in the next chapter.

Also, in the present chapter, the main applications of adaptive filters are revisited and discussed in greater detail.

2.2 Signal Representation

In this section, we briefly review some concepts related to deterministic and random discrete-time signals. Only specific results essential to the understanding of adaptive filtering are reviewed. For further details on signals and digital signal processing we refer to [6–13].

2.2.1 *Deterministic Signals*

A deterministic discrete-time signal is characterized by a defined mathematical function of the time index k,[1] with $k = 0, \pm 1, \pm 2, \pm 3, \ldots$. An example of a deterministic signal (or sequence) is

$$x(k) = \mathrm{e}^{-\alpha\, k} \cos(\omega k) + u(k) \tag{2.1}$$

where $u(k)$ is the unit step sequence.

The response of a linear time-invariant filter to an input $x(k)$ is given by the convolution summation, as follows [7]:

$$\begin{aligned} y(k) = x(k) * h(k) &= \sum_{n=-\infty}^{\infty} x(n)h(k-n) \\ &= \sum_{n=-\infty}^{\infty} h(n)x(k-n) = h(k) * x(k) \end{aligned} \tag{2.2}$$

where $h(k)$ is the impulse response of the filter.[2]

The $\mathcal{Z}$-transform of a given sequence $x(k)$ is defined as

$$\mathcal{Z}\{x(k)\} = X(z) = \sum_{k=-\infty}^{\infty} x(k)z^{-k} \tag{2.3}$$

[1] The index k can also denote space in some applications.

[2] An alternative and more accurate notation for the convolution summation would be $(x * h)(k)$ instead of $x(k) * h(k)$, since in the latter the index k appears twice whereas the resulting convolution is simply a function of k. We will keep the latter notation since it is more widely used.

for regions in the $\mathcal{Z}$-plane such that this summation converges. If the $\mathcal{Z}$-transform is defined for a given region of the $\mathcal{Z}$-plane, in other words the above summation converges in that region, the convolution operation can be replaced by a product of the $\mathcal{Z}$-transforms as follows [7]:

$$Y(z) = H(z)\, X(z) \tag{2.4}$$

where $Y(z)$, $X(z)$, and $H(z)$ are the $\mathcal{Z}$-transforms of $y(k)$, $x(k)$, and $h(k)$, respectively. Considering only waveforms that start at an instant $k \geq 0$ and have finite power, their $\mathcal{Z}$-transforms will always be defined outside the unit circle.

For finite-energy waveforms, it is convenient to use the discrete-time Fourier transform defined as

$$\mathcal{F}\{x(k)\} = X(\mathrm{e}^{\jmath\omega}) = \sum_{k=-\infty}^{\infty} x(k)\mathrm{e}^{-\jmath\omega k} \tag{2.5}$$

Although the discrete-time Fourier transform does not exist for a signal with infinite energy, if the signal has finite power, a generalized discrete-time Fourier transform exists and is largely used for deterministic signals [14].

2.2.2 *Random Signals*

A random variable X is a function that assigns a number to every outcome, denoted by ϱ, of a given experiment. A stochastic process is a rule to describe the time evolution of the random variable depending on ϱ, therefore it is a function of two variables $\mathrm{X}(k, \varrho)$. The set of all experimental outcomes, i.e., the ensemble, is the domain of ϱ. We denote $x(k)$ as a sample of the given process with ϱ fixed, where in this case if k is also fixed, $x(k)$ is a number. When any statistical operator is applied to $x(k)$ it is implied that $x(k)$ is a random variable, k is fixed, and ϱ is variable. In this book, $x(k)$ represents a random signal.

Random signals do not have a precise description of their waveforms. What is possible is to characterize them via measured statistics or through a probabilistic model. For random signals, the first- and second-order statistics are most of the time sufficient for characterization of the stochastic process. The first- and second-order statistics are also convenient for measurements. In addition, the effect on these statistics caused by linear filtering can be easily accounted for as shown below.

Let's consider for the time being that the random signals are real. We start to introduce some tools to deal with random signals by defining the distribution function of a random variable as

$$P_{x(k)}(y) \stackrel{\triangle}{=} \textit{probability of } x(k) \textit{ being smaller or equal to } y$$

or

$$P_{x(k)}(y) = \int_{-\infty}^{y} p_{x(k)}(z)dz \tag{2.6}$$

The derivative of the distribution function is the probability density function (pdf)

$$p_{x(k)}(y) = \frac{dP_{x(k)}(y)}{dy} \tag{2.7}$$

The expected value, or mean value, of the process is defined by

$$m_x(k) = E[x(k)] \tag{2.8}$$

The definition of the expected value is expressed as

$$E[x(k)] = \int_{-\infty}^{\infty} y \ p_{x(k)}(y)dy \tag{2.9}$$

where $p_{x(k)}(y)$ is the pdf of $x(k)$ at the point y.

The autocorrelation function of the process $x(k)$ is defined by

$$r_x(k,l) = E[x(k)x(l)] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} yzp_{x(k),x(l)}(y,z)dydz \tag{2.10}$$

where $p_{x(k),x(l)}(y,z)$ is the joint probability density of the random variables $x(k)$ and $x(l)$ defined as

$$p_{x(k),x(l)}(y,z) = \frac{\partial^2 P_{x(k),x(l)}(y,z)}{\partial y \partial z} \tag{2.11}$$

where

$$P_{x(k),x(l)}(y,z) \triangleq probability\,of\ \{x(k) \leq y\ and\ x(l) \leq z\}$$

The autocovariance function is defined as

$$\sigma_x^2(k,l) = E\{[x(k) - m_x(k)][x(l) - m_x(l)]\} = r_x(k,l) - m_x(k)m_x(l) \tag{2.12}$$

where the second equality follows from the definitions of mean value and autocorrelation. For $k = l$, $\sigma_x^2(k,l) = \sigma_x^2(k)$ which is the variance of $x(k)$.

The most important specific example of probability density function is the Gaussian density function, also known as normal density function [15, 16]. The Gaussian pdf is defined by

$$p_{x(k)}(y) = \frac{1}{\sqrt{2\pi\sigma_x^2(k)}} \mathrm{e}^{-\frac{(y-m_x(k))^2}{2\sigma_x^2(k)}} \tag{2.13}$$

where $m_x(k)$ and $\sigma_x^2(k)$ are the mean and variance of $x(k)$, respectively.

One justification for the importance of the Gaussian distribution is the central limit theorem. Given a random variable x composed by the sum of n independent random variables x_i as follows:

$$x = \sum_{i=1}^{n} x_i \tag{2.14}$$

the central limit theorem states that under certain general conditions, the probability density function of x approaches a Gaussian density function for large n. The mean and variance of x are given, respectively, by

$$m_x = \sum_{i=1}^{n} m_{x_i} \tag{2.15}$$

$$\sigma_x^2 = \sum_{i=1}^{n} \sigma_{x_i}^2 \tag{2.16}$$

Considering that the values of the mean and variance of x can grow, define

$$x^{'} = \frac{x - m_x}{\sigma_x} \tag{2.17}$$

In this case, for $n \rightarrow \infty$ it follows that

$$p_{x'}(y) = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{y^2}{2}} \tag{2.18}$$

In a number of situations we require the calculation of conditional distributions, where the probability of a certain event to occur is calculated assuming that another event B has occurred. In this case, we define

$$\begin{aligned} P_{x(k)}(y|B) &= \frac{P(\{x(k) \leq y\} \cap B)}{P(B)} \\ &\triangleq \textit{probability of } \; x(k) \leq y \textit{ assuming } B \textit{ has occurred} \end{aligned} \tag{2.19}$$

This joint event consists of all outcomes $\varrho \in B$ such that $x(k) = x(k, \varrho) \leq y$.[3] The definition of the conditional mean is given by

$$m_{x|B}(k) = E[x(k)|B] = \int_{-\infty}^{\infty} y p_{x(k)}(y|B) dy \tag{2.20}$$

where $p_{x(k)}(y|B)$ is the pdf of $x(k)$ conditioned on B.

[3] Or equivalently, such that $\mathrm{X}(k, \varrho) \leq y$.

The conditional variance is defined as

$$\sigma^2_{x|B}(k) = E\{[x(k) - m_{x|B}(k)]^2|B\} = \int_{-\infty}^{\infty} [y - m_{x|B}(k)]^2 p_{x(k)}(y|B)dy \quad (2.21)$$

There are processes for which the mean and autocorrelation functions are shift (or time) invariant, i.e.,

$$m_x(k-i) = m_x(k) = E[x(k)] = m_x \quad (2.22)$$

$$r_x(k,i) = E[x(k-j)x(i-j)] = r_x(k-i) = r_x(l) \quad (2.23)$$

and as a consequence

$$\sigma^2_x(l) = r_x(l) - m^2_x \quad (2.24)$$

These processes are said to be wide-sense stationary (WSS). If the nth-order statistics of a process is shift invariant, the process is said to be nth-order stationary. Also if the process is nth-order stationary for any value of n, the process is stationary in strict sense.

Two processes are considered jointly WSS if and only if any linear combination of them is also WSS. This is equivalent to state that

$$y(k) = k_1\, x_1(k) + k_2\, x_2(k) \quad (2.25)$$

must be WSS, for any constants k_1 and k_2, if $x_1(k)$ and $x_2(k)$ are jointly WSS. This property implies that both $x_1(k)$ and $x_2(k)$ have shift-invariant means and autocorrelations, and that their cross-correlation is also shift invariant.

For complex signals where $x(k) = x_r(k) + \jmath x_i(k)$, $y = y_r + \jmath y_i$, and $z = z_r + \jmath z_i$, we have the following definition of the expected value

$$E[x(k)] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} y p_{x_r(k),x_i(k)}(y_r, y_i)dy_r dy_i \quad (2.26)$$

where $p_{x_r(k),x_i(k)}(y_r, y_i)$ is the joint probability density function (pdf) of $x_r(k)$ and $x_i(k)$.

The autocorrelation function of the complex random signal $x(k)$ is defined by

$$\begin{aligned} r_x(k,l) &= E[x(k)x^*(l)] \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} yz^* p_{x_r(k),x_i(k),x_r(l),x_i(l)}(y_r, y_i, z_r, z_i)dy_r dy_i dz_r dz_i \end{aligned} \quad (2.27)$$

where $*$ denotes complex conjugate, since we assume for now that we are dealing with complex signals, and $p_{x_r(k),x_i(k),x_r(l),x_i(l)}(y_r, y_i, z_r, z_i)$ is the joint probability density function of the random variables $x(k)$ and $x(l)$.

For complex signals the autocovariance function is defined as

$$\sigma_x^2(k,l) = E\{[x(k) - m_x(k)][x(l) - m_x(l)]^*\} = r_x(k,l) - m_x(k)m_x^*(l) \quad (2.28)$$

2.2.2.1 Autoregressive Moving Average Process

The process resulting from the output of a system described by a general linear difference equation given by

$$y(k) = \sum_{j=0}^{M} b_j x(k-j) + \sum_{i=1}^{N} a_i y(k-i) \quad (2.29)$$

where $x(k)$ is a white noise, is called autoregressive moving average (ARMA) process. The coefficients a_i and b_j are the parameters of the ARMA process. The output signal $y(k)$ is also said to be a colored noise since the autocorrelation function of $y(k)$ is nonzero for a lag different from zero, i.e., $r(l) \neq 0$ for some $l \neq 0$.

For the special case where $b_j = 0$ for $j = 1, 2, \ldots, M$, the resulting process is called autoregressive (AR) process. The terminology means that the process depends on the present value of the input signal and on a linear combination of past samples of the process. This indicates the presence of a feedback of the output signal.

For the special case where $a_i = 0$ for $i = 1, 2, \ldots, N$, the process is identified as a moving average (MA) process. This terminology indicates that the process depends on a linear combination of the present and past samples of the input signal. In summary, an ARMA process can be generated by applying a white noise to the input of a digital filter with poles and zeros, whereas for the AR and MA cases the digital filters are all-pole and all-zero filters, respectively.

2.2.2.2 Markov Process

A stochastic process is called a Markov process if its past has no influence in the future when the present is specified [14, 15]. In other words, the present behavior of the process depends only on the most recent past, i.e., all behavior previous to the most recent past is not required. A first-order AR process is a first-order Markov process, whereas an Nth-order AR process is considered an Nth-order Markov process. Take as an example the sequence

$$y(k) = ay(k-1) + n(k) \quad (2.30)$$

where $n(k)$ is a white-noise process. The process represented by $y(k)$ is determined by $y(k-1)$ and $n(k)$, and no information before the instant $k-1$ is required. We conclude that $y(k)$ represents a Markov process. In the previous example, if $a = 1$ and $y(-1) = 0$ the signal $y(k)$, for $k \geq 0$, is a sum of white noise samples, usually called random walk sequence.

Formally, an mth-order Markov process satisfies the following condition: for all $k \geq 0$, and for a fixed m, it follows that

$$\begin{aligned} &P_{x(k)}\left(y|x(k-1), x(k-2), \ldots, x(0)\right) \\ &\quad = P_{x(k)}\left(y|x(k-1), x(k-2), \ldots, x(k-m)\right) \end{aligned} \tag{2.31}$$

2.2.2.3 Wold Decomposition

Another important result related to any WSS process $x(k)$ is the Wold decomposition, which states that $x(k)$ can be decomposed as

$$x(k) = x_r(k) + x_p(k) \tag{2.32}$$

where $x_r(k)$ is a regular process that is equivalent to the response of a stable, linear, time-invariant, and causal filter to a white noise [14], and $x_p(k)$ is a perfectly predictable (deterministic or singular) process. Also, $x_p(k)$ and $x_r(k)$ are orthogonal processes, i.e., $E[x_r(k)x_p(k)] = 0$. The key factor here is that the regular process can be modeled through a stable autoregressive model [17] with a stable and causal inverse. The importance of Wold decomposition lies on the observation that a WSS process can in part be represented by an AR process of adequate order, with the remaining part consisting of a perfectly predictable process. Obviously, the perfectly predictable process part of $x(k)$ also admits an AR model with zero excitation.

2.2.2.4 Power Spectral Density

Stochastic signals that are WSS are persistent and therefore are not finite-energy signals. On the other hand, they have finite power such that the generalized discrete-time Fourier transform can be applied to them. When the generalized discrete-time Fourier transform is applied to a WSS process it leads to a random function of the frequency [14]. On the other hand, the autocorrelation functions of most practical stationary processes have discrete-time Fourier transform. Therefore, the discrete-time Fourier transform of the autocorrelation function of a stationary random process can be very useful in many situations. This transform, called power spectral density, is defined as

$$R_x(\mathrm{e}^{\jmath\omega}) = \sum_{l=-\infty}^{\infty} r_x(l)\mathrm{e}^{-\jmath\omega l} = \mathcal{F}[r_x(l)] \tag{2.33}$$

where $r_x(l)$ is the autocorrelation of the process represented by $x(k)$. The inverse discrete-time Fourier transform allows us to recover $r_x(l)$ from $R_x(e^{\jmath\omega})$, through the relation

$$r_x(l) = \frac{1}{2\pi}\int_{-\pi}^{\pi} R_x(e^{\jmath\omega})e^{\jmath\omega l}d\omega = \mathcal{F}^{-1}[R_x(e^{\jmath\omega})] \tag{2.34}$$

It should be mentioned that $R_x(e^{\jmath\omega})$ is a deterministic function of ω and can be interpreted as the power density of the random process at a given frequency in the ensemble,[4] i.e., considering the average outcome of all possible realizations of the process. In particular, the mean squared value of the process represented by $x(k)$ is given by

$$r_x(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} R_x(e^{\jmath\omega})d\omega \tag{2.35}$$

If the random signal representing any single realization of a stationary process is applied as input to a linear and time-invariant filter, with impulse response $h(k)$, the following equalities are valid and can be easily verified:

$$y(k) = \sum_{n=-\infty}^{\infty} x(n)h(k-n) = x(k) * h(k) \tag{2.36}$$

$$r_y(l) = r_x(l) * r_h(l) \tag{2.37}$$

$$R_y(e^{\jmath\omega}) = R_x(e^{\jmath\omega})|H(e^{\jmath\omega})|^2 \tag{2.38}$$

$$r_{yx}(l) = r_x(l) * h(l) = E[x^*(k)y(k+l)] \tag{2.39}$$

$$R_{yx}(e^{\jmath\omega}) = R_x(e^{\jmath\omega})H(e^{\jmath\omega}) \tag{2.40}$$

where $r_h(l) = h(l) * h(-l)$, $R_y(e^{\jmath\omega})$ is the power spectral density of the output signal, $r_{yx}(k)$ is the cross-correlation of $x(k)$ and $y(k)$, and $R_{yx}(e^{\jmath\omega})$ is the corresponding cross-power spectral density.

The main feature of the spectral density function is to allow a simple analysis of the correlation behavior of WSS random signals processed with linear time-invariant systems. As an illustration, suppose a white noise is applied as input to a lowpass filter with impulse response $h(k)$ and sharp cutoff at a given frequency ω_l. The autocorrelation function of the output signal $y(k)$ will not be a single impulse, it will be $h(k) * h(-k)$. Therefore, the signal $y(k)$ will look like a band-limited random signal, in this case, a slow-varying noise. Some properties of the function $R_x(e^{\jmath\omega})$ of a discrete-time and stationary stochastic process are worth mentioning. The power spectrum density is a periodic function of ω, with period 2π, as can be verified from its definition. Also, since for a stationary and complex random process we

[4] The average signal power at a given sufficiently small frequency range, $\Delta\omega$, around a center frequency ω_0 is approximately given by $\frac{\Delta\omega}{2\pi}R_x(e^{\jmath\omega_0})$.

have $r_x(-l) = r_x^*(l)$, $R_x(\mathrm{e}^{\jmath\omega})$ is real. Despite the usefulness of the power spectrum density function in dealing with WSS processes, it will not be widely used in this book since usually the filters considered here are time varying. However, it should be noted its important role in areas such as spectrum estimation [18, 19].

If the $\mathcal{Z}$-transforms of the autocorrelation and cross-correlation functions exist, we can generalize the definition of power spectral density. In particular, the definition of (2.33) corresponds to the following relation

$$\mathcal{Z}[r_x(k)] = R_x(z) = \sum_{k=-\infty}^{\infty} r_x(k) z^{-k} \tag{2.41}$$

If the random signal representing any single realization of a stationary process is applied as input to a linear and time-invariant filter with impulse response $h(k)$, the following equalities are valid:

$$R_y(z) = R_x(z)H(z)H(z^{-1}) \tag{2.42}$$

and

$$R_{yx}(z) = R_x(z)H(z) \tag{2.43}$$

where $H(z) = \mathcal{Z}[h(l)]$. If we wish to calculate the cross-correlation of $y(k)$ and $x(k)$, namely $r_{yx}(0)$, we can use the inverse $\mathcal{Z}$-transform formula as follows:

$$\begin{aligned} E[y(k)x^*(k)] &= \frac{1}{2\pi\jmath} \oint R_{yx}(z) \frac{dz}{z} \\ &= \frac{1}{2\pi\jmath} \oint H(z) R_x(z) \frac{dz}{z} \end{aligned} \tag{2.44}$$

where the integration path is a counterclockwise closed contour in the region of convergence of $R_{yx}(z)$. The contour integral above equation is usually solved through the Cauchy's residue theorem [8].

2.2.3 Ergodicity

In the probabilistic approach, the statistical parameters of the real data are obtained through ensemble averages (or expected values). The estimation of any parameter of the stochastic process can be obtained by averaging a large number of realizations of the given process, at each instant of time. However, in many applications only a few or even a single sample of the process is available. In these situations, we need to find out in which cases the statistical parameters of the process can be estimated by using time average of a single sample (or ensemble member) of the process. This is obviously not possible if the desired parameter is time varying. The equivalence between the ensemble average and time average is called ergodicity [14, 15].

The time average of a given stationary process represented by $x(k)$ is calculated by

$$\hat{m}_{x_N} = \frac{1}{2N+1} \sum_{k=-N}^{N} x(k) \tag{2.45}$$

If

$$\sigma^2_{\hat{m}_{x_N}} = \lim_{N \to \infty} E\{|\hat{m}_{x_N} - m_x|^2\} = 0$$

the process is said to be mean-ergodic in the mean-square sense. Therefore, the mean-ergodic process has time average that approximates the ensemble average as $N \to \infty$. Obviously, $\hat{m}_{x_N}$ is an unbiased estimate of m_x since

$$E[\hat{m}_{x_N}] = \frac{1}{2N+1} \sum_{k=-N}^{N} E[x(k)] = m_x \tag{2.46}$$

Therefore, the process will be considered *ergodic* if the variance of $\hat{m}_{x_N}$ tends to zero ($\sigma^2_{\hat{m}_{x_N}} \to 0$) when $N \to \infty$. The variance $\sigma^2_{\hat{m}_{x_N}}$ can be expressed after some manipulations as

$$\sigma^2_{\hat{m}_{x_N}} = \frac{1}{2N+1} \sum_{l=-2N}^{2N} \sigma_x^2(k+l,k) \left(1 - \frac{|l|}{2N+1}\right) \tag{2.47}$$

where $\sigma_x^2(k+l,k)$ is the autocovariance of the stochastic process $x(k)$. The variance of $\hat{m}_{x_N}$ tends to zero if and only if

$$\lim_{N \to \infty} \frac{1}{N} \sum_{l=0}^{N} \sigma_x^2(k+l,k) \to 0$$

The above condition is necessary and sufficient to guarantee that the process is mean-ergodic.

The ergodicity concept can be extended to higher order statistics. In particular, for second-order statistics we can define the process

$$x_l(k) = x(k+l)x^*(k) \tag{2.48}$$

where the mean of this process corresponds to the autocorrelation of $x(k)$, i.e., $r_x(l)$. Mean-ergodicity of $x_l(k)$ implies mean-square ergodicity of the autocorrelation of $x(k)$.

The time average of $x_l(k)$ is given by

$$\hat{m}_{x_{l,N}} = \frac{1}{2N+1} \sum_{k=-N}^{N} x_l(k) \tag{2.49}$$

that is an unbiased estimate of $r_x(l)$. If the variance of $\hat{m}_{x_{l,N}}$ tends to zero as N tends to infinity, the process $x(k)$ is said to be mean-square ergodic of the autocorrelation, i.e.,

$$\lim_{N\to\infty} E\{|\hat{m}_{x_{l,N}} - r_x(l)|^2\} = 0 \tag{2.50}$$

The above condition is satisfied if and only if

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=0}^{N} E\{x(k+l)x^*(k)x(k+l+i)x^*(k+i)\} - r_x^2(l) = 0 \tag{2.51}$$

where it is assumed that $x(n)$ has stationary fourth-order moments. The concept of ergodicity can be extended to nonstationary processes [14], however, that is beyond the scope of this book.

2.3 The Correlation Matrix

Usually, adaptive filters utilize the available input signals at instant k in their updating equations. These inputs are the elements of the input signal vector denoted by

$$\mathbf{x}(k) = [x_0(k)\, x_1(k) \ldots x_N(k)]^T$$

The correlation matrix is defined as $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^H(k)]$, where $\mathbf{x}^H(k)$ is the Hermitian transposition of $\mathbf{x}(k)$, that means transposition followed by complex conjugation or vice versa. As will be noted, the characteristics of the correlation matrix play a key role in the understanding of properties of most adaptive-filtering algorithms. As a consequence, it is important to examine the main properties of the matrix $\mathbf{R}$. Some properties of the correlation matrix come from the statistical nature of the adaptive-filtering problem, whereas other properties derive from the linear algebra theory.

For a given input vector, the correlation matrix is given by

$$\begin{aligned}\mathbf{R} &= \begin{bmatrix} E[|x_0(k)|^2] & E[x_0(k)x_1^*(k)] & \cdots & E[x_0(k)x_N^*(k)] \\ E[x_1(k)x_0^*(k)] & E[|x_1(k)|^2] & \cdots & E[x_1(k)x_N^*(k)] \\ \vdots & \vdots & \ddots & \vdots \\ E[x_N(k)x_0^*(k)] & E[x_N(k)x_1^*(k)] & \cdots & E[|x_N(k)|^2] \end{bmatrix} \\ &= E[\mathbf{x}(k)\mathbf{x}^H(k)] \end{aligned} \tag{2.52}$$

The main properties of the **R** matrix are listed below:

1. The matrix **R** is positive semidefinite.

   Proof. Given an arbitrary complex weight vector **w**, we can form a signal given by

   $$y(k) = \mathbf{w}^H \mathbf{x}(k)$$

   The magnitude squared of $y(k)$ is

   $$y(k)y^*(k) = |y(k)|^2 = \mathbf{w}^H \mathbf{x}(k)\mathbf{x}^H(k)\mathbf{w} \geq 0$$

   The mean-square (MS) value of $y(k)$ is then given by

   $$\text{MS}[y(k)] = E[|y(k)|^2] = \mathbf{w}^H E[\mathbf{x}(k)\mathbf{x}^H(k)]\mathbf{w} = \mathbf{w}^H \mathbf{R}\mathbf{w} \geq 0$$

   Therefore, the matrix **R** is positive semidefinite. □

   Usually, the matrix **R** is positive definite, unless the signals that compose the input vector are linearly dependent. Linear-dependent signals are rarely found in practice.
2. The matrix **R** is Hermitian, i.e.,

   $$\mathbf{R} = \mathbf{R}^H \tag{2.53}$$

   Proof.

   $$\mathbf{R}^H = E\{[\mathbf{x}(k)\mathbf{x}^H(k)]^H\} = E[\mathbf{x}(k)\mathbf{x}^H(k)] = \mathbf{R} \qquad \square$$

3. A matrix is Toeplitz if the elements of the main diagonal and of any secondary diagonal are equal. When the input signal vector is composed of delayed versions of the same signal (i.e., $x_i(k) = x_0(k-i)$, for $i = 1, 2, \ldots, N$) taken from a WSS process, matrix **R** is Toeplitz.

Proof. For the delayed signal input vector, with $x(k)$ WSS, matrix $\mathbf{R}$ has the following form

$$\mathbf{R} = \begin{bmatrix} r_x(0) & r_x(1) & \cdots & r_x(N) \\ r_x(-1) & r_x(0) & \cdots & r_x(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ r_x(-N) & r_x(-N+1) & \cdots & r_x(0) \end{bmatrix} \tag{2.54}$$

By examining the right-hand side of the above equation, we can easily conclude that $\mathbf{R}$ is Toeplitz. □

Note that $r_x^*(i) = r_x(-i)$, what also follows from the fact that the matrix $\mathbf{R}$ is Hermitian.

If matrix $\mathbf{R}$ given by (2.54) is nonsingular for a given N, the input signal is said to be *persistently exciting* of order $N+1$. This means that the power spectral density $R_x(\mathrm{e}^{J\omega})$ is different from zero at least at $N+1$ points in the interval $0 < \omega \leq 2\pi$. It also means that a nontrivial Nth-order FIR filter (with at least one nonzero coefficient) cannot filter $x(k)$ to zero. Note that a nontrivial filter, with $x(k)$ as input, would require at least $N+1$ zeros in order to generate an output with all samples equal to zero. The absence of persistence of excitation implies the misbehavior of some adaptive algorithms [20, 21]. The definition of persistence of excitation is not unique, and it is algorithm dependent (see the book by Johnson [20] for further details).

From now on in this section, we discuss some properties of the correlation matrix related to its eigenvalues and eigenvectors. A number λ is an eigenvalue of the matrix $\mathbf{R}$, with a corresponding eigenvector $\mathbf{q}$, if and only if

$$\mathbf{R}\mathbf{q} = \lambda\mathbf{q} \tag{2.55}$$

or equivalently

$$\det(\mathbf{R} - \lambda\mathbf{I}) = 0 \tag{2.56}$$

where $\mathbf{I}$ is the $(N+1)$ by $(N+1)$ identity matrix. Equation (2.56) is called characteristic equation of $\mathbf{R}$ and has $(N+1)$ solutions for λ. We denote the $(N+1)$ eigenvalues of $\mathbf{R}$ by $\lambda_0, \lambda_1, \ldots, \lambda_N$. Note also that for every value of λ, the vector $\mathbf{q} = \mathbf{0}$ satisfies (2.55); however, we consider only those particular values of λ that are linked to a nonzero eigenvector $\mathbf{q}$.

Some important properties related to the eigenvalues and eigenvectors of $\mathbf{R}$, which will be useful in the following chapters, are listed below.

1. The eigenvalues of $\mathbf{R}^m$ are λ_i^m, for $i = 0, 1, 2, \ldots, N$.

   Proof. By premultiplying (2.55) by $\mathbf{R}^{m-1}$, we obtain

$$\begin{aligned} \mathbf{R}^{m-1}\mathbf{R}\mathbf{q}_i &= \mathbf{R}^{m-1}\lambda_i\mathbf{q}_i = \lambda_i\mathbf{R}^{m-2}\mathbf{R}\mathbf{q}_i \\ &= \lambda_i\mathbf{R}^{m-2}\lambda_i\mathbf{q}_i = \lambda_i^2\mathbf{R}^{m-3}\mathbf{R}\mathbf{q}_i \\ &= \cdots = \lambda_i^m\mathbf{q}_i \end{aligned} \tag{2.57}$$

□

2. Suppose $\mathbf{R}$ has $N + 1$ linearly independent eigenvectors $\mathbf{q}_i$; then if we form a matrix $\mathbf{Q}$ with columns consisting of the $\mathbf{q}_i$'s, it follows that

$$\mathbf{Q}^{-1}\mathbf{R}\mathbf{Q} = \begin{bmatrix} \lambda_0 & 0 & \cdots & 0 \\ 0 & \lambda_1 & & \vdots \\ \vdots & 0 & \cdots & \vdots \\ \vdots & \vdots & & 0 \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} = \boldsymbol{\Lambda} \tag{2.58}$$

Proof.

$$\begin{aligned} \mathbf{R}\mathbf{Q} &= \mathbf{R}[\mathbf{q}_0\ \mathbf{q}_1 \cdots \mathbf{q}_N] = [\lambda_0\mathbf{q}_0\ \lambda_1\mathbf{q}_1 \cdots \lambda_N\mathbf{q}_N] \\ &= \mathbf{Q}\begin{bmatrix} \lambda_0 & 0 & \cdots & 0 \\ 0 & \lambda_1 & & \vdots \\ \vdots & 0 & \cdots & \vdots \\ \vdots & \vdots & & 0 \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} = \mathbf{Q}\boldsymbol{\Lambda} \end{aligned}$$

Therefore, since $\mathbf{Q}$ is invertible because the $\mathbf{q}_i$'s are linearly independent, we can show that

$$\mathbf{Q}^{-1}\mathbf{R}\mathbf{Q} = \boldsymbol{\Lambda}$$

□

3. The nonzero eigenvectors $\mathbf{q}_0, \mathbf{q}_1, \ldots \mathbf{q}_N$ that correspond to different eigenvalues are linearly independent.

Proof. If we form a linear combination of the eigenvectors such that

$$a_0\mathbf{q}_0 + a_1\mathbf{q}_1 + \cdots + a_N\mathbf{q}_N = \mathbf{0} \tag{2.59}$$

By multiplying the above equation by $\mathbf{R}$ we have

$$a_0\mathbf{R}\mathbf{q}_0 + a_1\mathbf{R}\mathbf{q}_1 + \cdots + a_N\mathbf{R}\mathbf{q}_N = a_0\lambda_0\mathbf{q}_0 + a_1\lambda_1\mathbf{q}_1 + \cdots + a_N\lambda_N\mathbf{q}_N = \mathbf{0} \tag{2.60}$$

Now by multiplying (2.59) by λ_N and subtracting the result from (2.60), we obtain

$$a_0(\lambda_0 - \lambda_N)\mathbf{q}_0 + a_1(\lambda_1 - \lambda_N)\mathbf{q}_1 + \cdots + a_{N-1}(\lambda_{N-1} - \lambda_N)\mathbf{q}_{N-1} = \mathbf{0}$$

By repeating the above steps, i.e., multiplying the above equation by $\mathbf{R}$ in one instance and by λ_{N-1} on the other instance, and subtracting the results, it yields

$$a_0(\lambda_0 - \lambda_N)(\lambda_0 - \lambda_{N-1})\mathbf{q}_0 + a_1(\lambda_1 - \lambda_N)(\lambda_1 - \lambda_{N-1})\mathbf{q}_1$$
$$+ \cdots + a_{N-2}(\lambda_{N-2} - \lambda_{N-1})\mathbf{q}_{N-2} = \mathbf{0}$$

By repeating the same above steps several times, we end up with

$$a_0(\lambda_0 - \lambda_N)(\lambda_0 - \lambda_{N-1}) \cdots (\lambda_0 - \lambda_1)\mathbf{q}_0 = \mathbf{0}$$

Since we assumed $\lambda_0 \neq \lambda_1$, $\lambda_0 \neq \lambda_2$, $\ldots$ $\lambda_0 \neq \lambda_N$, and $\mathbf{q}_0$ was assumed nonzero, then $a_0 = 0$.

The same line of thought can be used to show that $a_0 = a_1 = a_2 = \cdots = a_N = 0$ is the only solution for (2.59). Therefore, the eigenvectors corresponding to different eigenvalues are linearly independent. □

Not all matrices are diagonalizable. A matrix of order $(N+1)$ is diagonalizable if it possesses $(N+1)$ linearly independent eigenvectors. A matrix with repeated eigenvalues can be diagonalized or not, depending on the linear dependency of the eigenvectors. A nondiagonalizable matrix is called defective [22].

4. Since the correlation matrix $\mathbf{R}$ is Hermitian, i.e., $\mathbf{R}^H = \mathbf{R}$, its eigenvalues are real. These eigenvalues are equal to or greater than zero given that $\mathbf{R}$ is positive semidefinite.

Proof. First note that given an arbitrary complex vector $\mathbf{w}$,

$$(\mathbf{w}^H \mathbf{R} \mathbf{w})^H = \mathbf{w}^H \mathbf{R}^H (\mathbf{w}^H)^H = \mathbf{w}^H \mathbf{R} \mathbf{w}$$

Therefore, $\mathbf{w}^H \mathbf{R} \mathbf{w}$ is a real number. Assume now that λ_i is an eigenvalue of $\mathbf{R}$ corresponding to the eigenvector $\mathbf{q}_i$, i.e., $\mathbf{R}\mathbf{q}_i = \lambda_i \mathbf{q}_i$. By premultiplying this equation by $\mathbf{q}_i^H$, it follows that

$$\mathbf{q}_i^H \mathbf{R} \mathbf{q}_i = \lambda_i \mathbf{q}_i^H \mathbf{q}_i = \lambda_i \|\mathbf{q}_i\|^2$$

where the operation $\|\mathbf{a}\|^2 = |a_0|^2 + |a_1|^2 + \cdots + |a_N|^2$ is the Euclidean norm squared of the vector $\mathbf{a}$, that is always real. Since the term on the left hand is also real, $\|\mathbf{q}_i\|^2 \neq 0$, and $\mathbf{R}$ is positive semidefinite, we can conclude that λ_i is real and nonnegative. □

Note that $\mathbf{Q}$ is not unique since each $\mathbf{q}_i$ can be multiplied by an arbitrary nonzero constant, and the resulting vector continues to be an eigenvector.[5] For practical reasons, we consider only normalized eigenvectors having length one, that is

$$\mathbf{q}_i^H \mathbf{q}_i = 1 \quad \text{for } i = 0, 1, \ldots, N \tag{2.61}$$

[5] We can also change the order in which the $\mathbf{q}_i$'s compose matrix $\mathbf{Q}$, but this fact is not relevant to the present discussion.

5. If $\mathbf{R}$ is a Hermitian matrix with different eigenvalues, the eigenvectors are orthogonal to each other. As a consequence, there is a diagonalizing matrix $\mathbf{Q}$ that is unitary, i.e., $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$.

   Proof. Given two eigenvalues λ_i and λ_j, it follows that

$$\mathbf{R}\mathbf{q}_i = \lambda_i \mathbf{q}_i$$

   and

$$\mathbf{R}\mathbf{q}_j = \lambda_j \mathbf{q}_j \tag{2.62}$$

   Using the fact that $\mathbf{R}$ is Hermitian and that λ_i and λ_j are real, then

$$\mathbf{q}_i^H \mathbf{R} = \lambda_i \mathbf{q}_i^H$$

   and by multiplying this equation on the right by $\mathbf{q}_j$, we get

$$\mathbf{q}_i^H \mathbf{R}\mathbf{q}_j = \lambda_i \mathbf{q}_i^H \mathbf{q}_j$$

   Now by premultiplying (2.62) by $\mathbf{q}_i^H$, it follows that

$$\mathbf{q}_i^H \mathbf{R}\mathbf{q}_j = \lambda_j \mathbf{q}_i^H \mathbf{q}_j$$

   Therefore,

$$\lambda_i \mathbf{q}_i^H \mathbf{q}_j = \lambda_j \mathbf{q}_i^H \mathbf{q}_j$$

   Since $\lambda_i \neq \lambda_j$, it can be concluded that

$$\mathbf{q}_i^H \mathbf{q}_j = 0 \quad \text{for } i \neq j$$

   If we form matrix $\mathbf{Q}$ with normalized eigenvectors, matrix $\mathbf{Q}$ is a unitary matrix. □

   An important result is that any Hermitian matrix $\mathbf{R}$ can be diagonalized by a suitable unitary matrix $\mathbf{Q}$, even if the eigenvalues of $\mathbf{R}$ are not distinct. The proof is omitted here and can be found in [22]. Therefore, for Hermitian matrices with repeated eigenvalues it is always possible to find a complete set of orthonormal eigenvectors.

   A useful form to decompose a Hermitian matrix that results from the last property is

$$\mathbf{R} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^H = \sum_{i=0}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^H \tag{2.63}$$

   that is known as *spectral decomposition*. From this decomposition, one can easily derive the following relation

$$\mathbf{w}^H\mathbf{R}\mathbf{w} = \sum_{i=0}^{N} \lambda_i \mathbf{w}^H \mathbf{q}_i \mathbf{q}_i^H \mathbf{w} = \sum_{i=0}^{N} \lambda_i |\mathbf{w}^H \mathbf{q}_i|^2 \tag{2.64}$$

In addition, since $\mathbf{q}_i = \lambda_i \mathbf{R}^{-1}\mathbf{q}_i$, the eigenvectors of a matrix and of its inverse coincide, whereas the eigenvalues are reciprocals of each other. As a consequence,

$$\mathbf{R}^{-1} = \sum_{i=0}^{N} \frac{1}{\lambda_i} \mathbf{q}_i \mathbf{q}_i^H \tag{2.65}$$

Another consequence of the unitary property of $\mathbf{Q}$ for Hermitian matrices is that any Hermitian matrix can be written in the form

$$\begin{aligned}\mathbf{R} &= \left[\sqrt{\lambda_0}\mathbf{q}_0 \ \sqrt{\lambda_1}\mathbf{q}_1 \ldots \sqrt{\lambda_N}\mathbf{q}_N\right] \begin{bmatrix} \sqrt{\lambda_0}\mathbf{q}_0^H \\ \sqrt{\lambda_1}\mathbf{q}_1^H \\ \vdots \\ \sqrt{\lambda_N}\mathbf{q}_N^H \end{bmatrix} \\ &= \mathbf{L}\mathbf{L}^H \end{aligned} \tag{2.66}$$

6. The sum of the eigenvalues of **R** is equal to the trace of **R**, and the product of the eigenvalues of **R** is equal to the determinant of **R**.[6]

   Proof.

   $$\text{tr}[\mathbf{Q}^{-1}\mathbf{R}\mathbf{Q}] = \text{tr}[\boldsymbol{\Lambda}]$$

   where, $\text{tr}[\mathbf{A}] = \sum_{i=0}^{N} a_{ii}$. Since $\text{tr}[\mathbf{A}'\mathbf{A}] = \text{tr}[\mathbf{A}\mathbf{A}']$, we have

   $$\text{tr}[\mathbf{Q}^{-1}\mathbf{R}\mathbf{Q}] = \text{tr}[\mathbf{R}\mathbf{Q}\mathbf{Q}^{-1}] = \text{tr}[\mathbf{R}\mathbf{I}] = \text{tr}[\mathbf{R}] = \sum_{i=0}^{N} \lambda_i$$

   Also

   $$\det[\mathbf{Q}^{-1}\ \mathbf{R}\ \mathbf{Q}] = \det[\mathbf{R}]\det[\mathbf{Q}]\det[\mathbf{Q}^{-1}] = \det[\mathbf{R}] = \det[\boldsymbol{\Lambda}] = \prod_{i=0}^{N} \lambda_i. \qquad \square$$

7. The Rayleigh's quotient defined as

   $$\mathcal{R} = \frac{\mathbf{w}^H\mathbf{R}\mathbf{w}}{\mathbf{w}^H\mathbf{w}} \tag{2.67}$$

   of a Hermitian matrix is bounded by the minimum and maximum eigenvalues, i.e.,

   $$\lambda_{\min} \leq \mathcal{R} \leq \lambda_{\max} \tag{2.68}$$

[6] This property is valid for any square matrix, but for more general matrices the proof differs from the one presented here.

where the minimum and maximum values are reached when the vector $\mathbf{w}$ is chosen to be the eigenvector corresponding to the minimum and maximum eigenvalues, respectively.

Proof. Suppose $\mathbf{w} = \mathbf{Q}\mathbf{w}'$, where $\mathbf{Q}$ is the matrix that diagonalizes $\mathbf{R}$, then

$$\begin{aligned}\mathcal{R} &= \frac{\mathbf{w}'^H \mathbf{Q}^H \mathbf{R}\mathbf{Q}\mathbf{w}'}{\mathbf{w}'^H \mathbf{Q}^H \mathbf{Q}\mathbf{w}'} \\ &= \frac{\mathbf{w}'^H \boldsymbol{\Lambda} \mathbf{w}'}{\mathbf{w}'^H \mathbf{w}'} \\ &= \frac{\sum_{i=0}^{N} \lambda_i w_i'^2}{\sum_{i=0}^{N} w_i'^2} \end{aligned} \qquad (2.69)$$

It is then possible to show, see Problem 14, that the minimum value for the above equation occurs when $w_i = 0$ for $i \neq j$ and λ_j is the smallest eigenvalue. Identically, the maximum value for $\mathcal{R}$ occurs when $w_i = 0$ for $i \neq l$, where λ_l is the largest eigenvalue. □

There are several ways to define the norm of a matrix. In this book the norm of a matrix $\mathbf{R}$, denoted by $\|\mathbf{R}\|$, is defined by

$$\begin{aligned}\|\mathbf{R}\|^2 &= \max_{\mathbf{w}\neq 0} \frac{\|\mathbf{R}\mathbf{w}\|^2}{\|\mathbf{w}\|^2} \\ &= \max_{\mathbf{w}\neq 0} \frac{\mathbf{w}^H \mathbf{R}^H \mathbf{R}\mathbf{w}}{\mathbf{w}^H \mathbf{w}} \end{aligned} \qquad (2.70)$$

Note that the norm of $\mathbf{R}$ is a measure of how a vector $\mathbf{w}$ grows in magnitude, when it is multiplied by $\mathbf{R}$.

When the matrix $\mathbf{R}$ is Hermitian, the norm of $\mathbf{R}$ is easily obtained by using the results of (2.57) and (2.68). The result is

$$\|\mathbf{R}\| = \lambda_{\max} \qquad (2.71)$$

where $\lambda_{\max}$ is the maximum eigenvalue of $\mathbf{R}$.

A common problem that we encounter in adaptive filtering is the solution of a system of linear equations such as

$$\mathbf{R}\mathbf{w} = \mathbf{p} \qquad (2.72)$$

In case there is an error in the vector $\mathbf{p}$, originated by quantization or estimation, how does it affect the solution of the system of linear equations? For a positive definite Hermitian matrix $\mathbf{R}$, it can be shown [22] that the relative error in the solution of the above linear system of equations is bounded by

$$\frac{\|\Delta \mathbf{w}\|}{\|\mathbf{w}\|} \le \frac{\lambda_{\max}}{\lambda_{\min}} \frac{\|\Delta \mathbf{p}\|}{\|\mathbf{p}\|} \tag{2.73}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the maximum and minimum values of the eigenvalues of $\mathbf{R}$, respectively. The ratio $\lambda_{\max}/\lambda_{\min}$ is called condition number of a matrix, that is

$$C = \frac{\lambda_{\max}}{\lambda_{\min}} = \|\mathbf{R}\| \|\mathbf{R}^{-1}\| \tag{2.74}$$

The value of C influences the convergence behavior of a number of adaptive-filtering algorithms, as will be seen in the following chapters. Large value of C indicates that the matrix $\mathbf{R}$ is ill-conditioned, and that errors introduced by the manipulation of $\mathbf{R}$ may be largely amplified. When $C = 1$, the matrix is perfectly conditioned. In case $\mathbf{R}$ represents the correlation matrix of the input signal of an adaptive filter, with the input vector composed by uncorrelated elements of a delay line (see Fig. 2.1b, and the discussions around it), then $C = 1$.

Example 2.1. Suppose the input signal vector is composed by a delay line with a single input signal, i.e.,

$$\mathbf{x}(k) = [x(k)\, x(k-1) \ldots x(k-N)]^T$$

Given the following input signals:

(a)

$$x(k) = n(k)$$

(b)

$$x(k) = a \cos \omega_0 k + n(k)$$

(c)

$$x(k) = \sum_{i=0}^{M} b_i n(k-i)$$

(d)

$$x(k) = -a_1 x(k-1) + n(k)$$

(e)

$$x(k) = a \mathrm{e}^{\jmath(\omega_0 k + n(k))}$$

where $n(k)$ is a white noise with zero mean and variance σ_n^2; in case (e) $n(k)$ is uniformly distributed in the range $-\pi$ to π.

Calculate the autocorrelation matrix $\mathbf{R}$ for $N = 3$.

Solution. (a) In this case, we have that $E[x(k)x(k-l)] = \sigma_n^2 \delta(l)$, where $\delta(l)$ denotes an impulse sequence. Therefore,

$$\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)] = \sigma_n^2 \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \tag{2.75}$$

(b) In this example, $n(k)$ is zero mean and uncorrelated with the deterministic cosine. The autocorrelation function can then be expressed as

$$\begin{aligned} r(k, k-l) &= E[a^2 \cos(\omega_0 k)\cos(\omega_0 k - \omega_0 l) + n(k)n(k-l)] \\ &= a^2 E[\cos(\omega_0 k)\cos(\omega_0 k - \omega_0 l)] + \sigma_n^2 \delta(l) \\ &= \frac{a^2}{2}[\cos(\omega_0 l) + \cos(2\omega_0 k - \omega_0 l)] + \sigma_n^2 \delta(l) \end{aligned} \tag{2.76}$$

where $\delta(l)$ again denotes an impulse sequence. Since part of the input signal is deterministic and nonstationary, the autocorrelation is time dependent.

For the 3×3 case the input signal correlation matrix $\mathbf{R}(k)$ becomes

$$\frac{a^2}{2}\begin{bmatrix} 1+\cos 2\omega_0 k + \frac{2}{a^2}\sigma_n^2 & \cos\omega_0 + \cos\omega_0(2k-1) & \cos 2\omega_0 + \cos 2\omega_0(k-1) \\ \cos\omega_0 + \cos\omega_0(2k-1) & 1+\cos 2\omega_0(k-1) + \frac{2}{a^2}\sigma_n^2 & \cos\omega_0 + \cos\omega_0(2(k-1)-1) \\ \cos 2\omega_0 + \cos 2\omega_0(k-1) & \cos\omega_0 + \cos\omega_0(2(k-1)-1) & 1+\cos 2\omega_0(k-2) + \frac{2}{a^2}\sigma_n^2 \end{bmatrix}$$

(c) By exploring the fact that $n(k)$ is a white noise, we can perform the following simplifications:

$$\begin{aligned} r(l) = E[x(k)x(k-l)] &= E\left[\sum_{j=0}^{M-l}\sum_{i=0}^{M} b_i b_j n(k-i)n(k-l-j)\right] \\ &= \sum_{j=0}^{M-l} b_j b_{l+j} E[n^2(k-l-j)] = \sigma_n^2 \sum_{j=0}^{M} b_j b_{l+j} \\ & 0 \le l+j \le M \end{aligned} \tag{2.77}$$

where from the third to the fourth relation we used the fact that $E[n(k-i)n(k-l-j)] = 0$ for $i \neq l+j$. For $M = 3$, the correlation matrix has the following form

$$\mathbf{R} = \sigma_n^2 \begin{bmatrix} \sum_{i=0}^{3} b_i^2 & \sum_{i=0}^{2} b_i b_{i+1} & \sum_{i=0}^{1} b_i b_{i+2} & b_0 b_3 \\ \sum_{i=0}^{2} b_i b_{i+1} & \sum_{i=0}^{3} b_i^2 & \sum_{i=0}^{2} b_i b_{i+1} & \sum_{i=0}^{1} b_i b_{i+2} \\ \sum_{i=0}^{1} b_i b_{i+2} & \sum_{i=0}^{2} b_i b_{i+1} & \sum_{i=0}^{3} b_i^2 & \sum_{i=0}^{2} b_i b_{i+1} \\ b_0 b_3 & \sum_{i=0}^{1} b_i b_{i+2} & \sum_{i=0}^{2} b_i b_{i+1} & \sum_{i=0}^{3} b_i^2 \end{bmatrix} \tag{2.78}$$

(d) By solving the difference equation, we can obtain the correlation between $x(k)$ and $x(k-l)$, that is

$$x(k) = (-a_1)^l x(k-l) + \sum_{j=0}^{l-1} (-a_1)^j n(k-j) \tag{2.79}$$

Multiplying $x(k-l)$ on both sides of the above equation and taking the expected value of the result, we obtain

$$E[x(k)x(k-l)] = (-a_1)^l E[x^2(k-l)] \tag{2.80}$$

since $x(k-l)$ is independent of $n(k-j)$ for $j \leq l-1$.

For $l = 0$, just calculate $x^2(k)$ and apply the expectation operation to the result. The partial result is

$$E[x^2(k)] = a_1^2 E[x^2(k-1)] + E[n^2(k)] \tag{2.81}$$

therefore,

$$E[x^2(k)] = \frac{\sigma_n^2}{1-a_1^2} \tag{2.82}$$

assuming $x(k)$ is WSS.

The elements of $\mathbf{R}$ are then given by

$$r(l) = \frac{(-a_1)^{|l|}}{1-a_1^2}\sigma_n^2 \tag{2.83}$$

and the 3 × 3 autocorrelation matrix becomes

$$\mathbf{R} = \frac{\sigma_n^2}{1-a_1^2}\begin{bmatrix} 1 & -a_1 & a_1^2 \\ -a_1 & 1 & -a_1 \\ a_1^2 & -a_1 & 1 \end{bmatrix}$$

(e) In this case, we are interested in calculating the autocorrelation of a complex sequence, that is

$$\begin{aligned} r(l) &= E[x(k)x^*(k-l)] \\ &= a^2 E[\mathrm{e}^{-\jmath(-\omega_0 l - n(k) + n(k-l))}] \end{aligned} \tag{2.84}$$

By recalling the definition of expected value in (2.9), for $l \neq 0$,

$$
\begin{aligned}
r(l) &= a^2 \mathrm{e}^{\jmath\omega_0 l} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathrm{e}^{-\jmath(-n_0+n_1)} p_{n(k),n(k-l)}(n_0, n_1) dn_0 dn_1 \\
&= a^2 \mathrm{e}^{\jmath\omega_0 l} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\jmath(-n_0+n_1)} p_{n(k)}(n_0) p_{n(k-l)}(n_1) dn_0 dn_1 \\
&= a^2 \mathrm{e}^{\jmath\omega_0 l} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\jmath(-n_0+n_1)} \frac{1}{2\pi} \frac{1}{2\pi} dn_0 dn_1 \\
&= a^2 \mathrm{e}^{\jmath\omega_0 l} \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\jmath(-n_0+n_1)} dn_0 dn_1 \\
&= a^2 \mathrm{e}^{\jmath\omega_0 l} \frac{1}{4\pi^2} \left[\int_{-\pi}^{\pi} \mathrm{e}^{\jmath n_0} dn_0 \right] \left[\int_{-\pi}^{\pi} \mathrm{e}^{-\jmath n_1} dn_1 \right] \\
&= a^2 \mathrm{e}^{\jmath\omega_0 l} \frac{1}{4\pi^2} \left[\frac{\mathrm{e}^{\jmath\pi} - \mathrm{e}^{-\jmath\pi}}{\jmath} \right] \left[\frac{-\mathrm{e}^{-\jmath\pi} + \mathrm{e}^{\jmath\pi}}{\jmath} \right] \\
&= -a^2 \mathrm{e}^{\jmath\omega_0 l} \frac{1}{\pi^2} (\sin\pi)(\sin\pi) = 0
\end{aligned}
\tag{2.85}
$$

where in the fifth equality it is used the fact that $n(k)$ and $n(k-l)$, for $l \neq 0$, are independent.

For $l = 0$

$$
r(0) = E[x(k)x^*(k)] = a^2 \mathrm{e}^{\jmath(\omega_0 0)} = a^2
$$

Therefore,

$$
r(l) = E[x(k)x^*(k-l)] = a^2 \mathrm{e}^{\jmath(\omega_0 l)} \delta(l)
$$

where in the 3 × 3 case

$$
\mathbf{R} = \begin{bmatrix} a^2 & 0 & 0 \\ 0 & a^2 & 0 \\ 0 & 0 & a^2 \end{bmatrix}
$$

At the end it was verified the fact that when we have two exponential functions ($l \neq 0$) with uniformly distributed white noise in the range of $-k\pi$ to $k\pi$ as exponents, these exponentials are nonorthogonal only if $l = 0$, where k is a positive integer. □

In the remaining part of this chapter and in the following chapters, we will treat the algorithms for real and complex signals separately. The derivations of the adaptive-filtering algorithms for complex signals are usually straightforward extensions of the real signal cases, and some of them are left as exercises.

2.4 Wiener Filter

One of the most widely used objective function in adaptive filtering is the MSE defined as

$$F[e(k)] = \xi(k) = E[e^2(k)] = E[d^2(k) - 2d(k)y(k) + y^2(k)] \qquad (2.86)$$

where $d(k)$ is the reference signal as illustrated in Fig. 1.1.

Suppose the adaptive filter consists of a linear combiner, i.e., the output signal is composed by a linear combination of signals coming from an array as depicted in Fig. 2.1a. In this case,

$$y(k) = \sum_{i=0}^{N} w_i(k)x_i(k) = \mathbf{w}^T(k)\mathbf{x}(k) \qquad (2.87)$$

where $\mathbf{x}(k) = [x_0(k)\; x_1(k) \ldots x_N(k)]^T$ and $\mathbf{w}(k) = [w_0(k)\; w_1(k) \ldots w_N(k)]^T$ are the input signal and the adaptive-filter coefficient vectors, respectively.

In many applications, each element of the input signal vector consists of a delayed version of the same signal, that is: $x_0(k) = x(k), x_1(k) = x(k-1), \ldots, x_N(k) = x(k-N)$. Note that in this case the signal $y(k)$ is the result of applying an FIR filter to the input signal $x(k)$.

Since most of the analyses and algorithms presented in this book apply equally to the linear combiner and the FIR filter cases, we will mostly consider the latter case throughout the rest of the book. The main reason for this decision is that the fast algorithms for the recursive least-squares solution, to be discussed in the forthcoming chapters, explore the fact that the input signal vector consists of the output of a delay line with a single input signal, and, as a consequence, are not applicable to the linear combiner case.

The most straightforward realization for the adaptive filter is through the direct-form FIR structure as illustrated in Fig. 2.1b, with the output given by

$$y(k) = \sum_{i=0}^{N} w_i(k)x(k-i) = \mathbf{w}^T(k)\mathbf{x}(k) \qquad (2.88)$$

where $\mathbf{x}(k) = [x(k)\; x(k-1) \ldots x(k-N)]^T$ is the input vector representing a tapped-delay line, and $\mathbf{w}(k) = [w_0(k)\; w_1(k) \ldots w_N(k)]^T$ is the tap-weight vector.

In both the linear combiner and FIR filter cases, the objective function can be rewritten as

$$\begin{aligned} E[e^2(k)] &= \xi(k) \\ &= E\left[d^2(k) - 2d(k)\mathbf{w}^T(k)\mathbf{x}(k) + \mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)\right] \\ &= E[d^2(k)] - 2E[d(k)\mathbf{w}^T(k)\mathbf{x}(k)] + E[\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)] \qquad (2.89) \end{aligned}$$

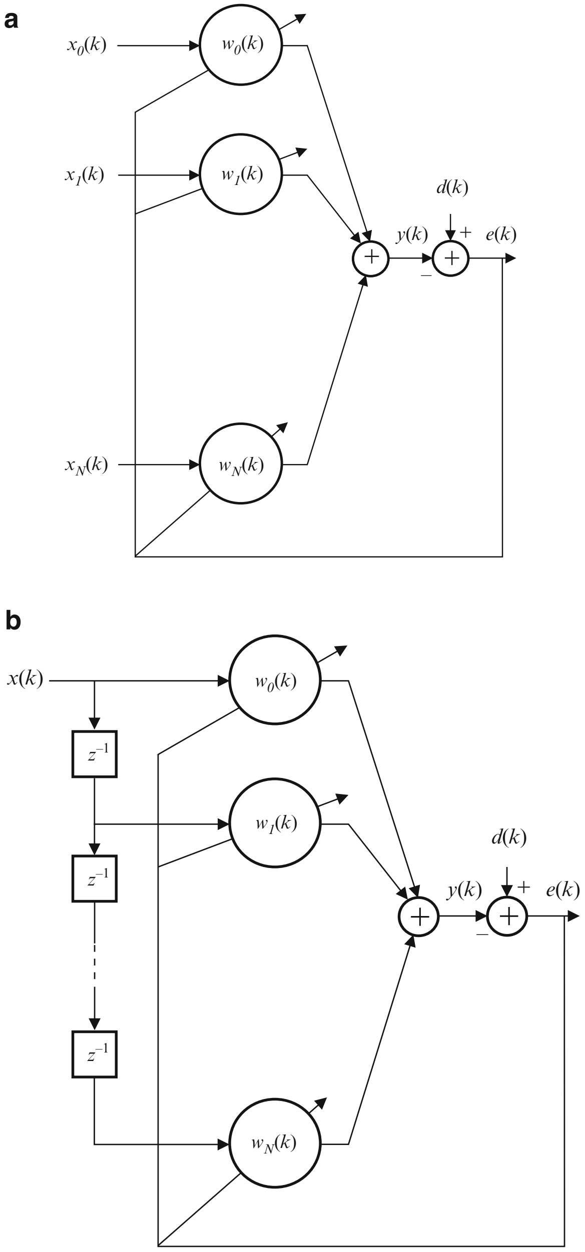

Fig. 2.1 (**a**) Linear combiner; (**b**) Adaptive FIR filter

For a filter with fixed coefficients, the MSE function in a stationary environment is given by

$$
\begin{aligned}
\xi &= E[d^2(k)] - 2\mathbf{w}^T E[d(k)\mathbf{x}(k)] + \mathbf{w}^T E[\mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w} \\
&= E[d^2(k)] - 2\mathbf{w}^T\mathbf{p} + \mathbf{w}^T\mathbf{R}\mathbf{w}
\end{aligned}
\tag{2.90}
$$

where $\mathbf{p} = E[d(k)\mathbf{x}(k)]$ is the cross-correlation vector between the desired and input signals, and $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$ is the input signal correlation matrix. As can be noted, the objective function ξ is a quadratic function of the tap-weight coefficients which would allow a straightforward solution for $\mathbf{w}$ that minimizes ξ, if vector $\mathbf{p}$ and matrix $\mathbf{R}$ are known. Note that matrix $\mathbf{R}$ corresponds to the Hessian matrix of the objective function defined in the previous chapter.

If the adaptive filter is implemented through an IIR filter, the objective function is a nonquadratic function of the filter parameters, turning the minimization problem into a much more difficult one. Local minima are likely to exist, rendering some solutions obtained by gradient-based algorithms unacceptable. Despite its disadvantages, adaptive IIR filters are needed in a number of applications where the order of a suitable FIR filter is too high. Typical applications include data equalization in communication channels and cancellation of acoustic echo, see Chap. 10.

The gradient vector of the MSE function related to the filter tap-weight coefficients is given by[7]

$$\begin{aligned}\mathbf{g_w} &= \frac{\partial \xi}{\partial \mathbf{w}} = \left[\frac{\partial \xi}{\partial w_0}\ \frac{\partial \xi}{\partial w_1} \cdots \frac{\partial \xi}{\partial w_N}\right]^T \\ &= -2\mathbf{p} + 2\mathbf{R}\mathbf{w} \qquad (2.91)\end{aligned}$$

By equating the gradient vector to zero and assuming $\mathbf{R}$ is nonsingular, the optimal values for the tap-weight coefficients that minimize the objective function can be evaluated as follows:

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} \qquad (2.92)$$

This solution is called the Wiener solution. Unfortunately, in practice, precise estimations of $\mathbf{R}$ and $\mathbf{p}$ are not available. When the input and the desired signals are ergodic, one is able to use time averages to estimate $\mathbf{R}$ and $\mathbf{p}$, what is implicitly performed by most adaptive algorithms.

If we replace the optimal solution for $\mathbf{w}$ in the MSE expression, we can calculate the minimum MSE provided by the Wiener solution:

$$\begin{aligned}\xi_{\min} &= E[d^2(k)] - 2\mathbf{w}_o^T\mathbf{p} + \mathbf{w}_o^T\mathbf{R}\mathbf{R}^{-1}\mathbf{p} \\ &= E[d^2(k)] - \mathbf{w}_o^T\mathbf{p} \qquad (2.93)\end{aligned}$$

The above equation indicates that the optimal set of parameters removes part of the power of the desired signal through the cross-correlation between $x(k)$ and $d(k)$, assuming both signals stationary. If the reference signal and the input signal are orthogonal, the optimal coefficients are equal to zero and the minimum MSE is

[7] Some books define $\mathbf{g_w}$ as $\left[\frac{\partial \xi}{\partial \mathbf{w}}\right]^T$, here we follow the notation more widely used in the subject matter.

$E[d^2(k)]$. This result is expected since nothing can be done with the parameters in order to minimize the MSE if the input signal carries no information about the desired signal. In this case, if any of the taps is nonzero, it would only increase the MSE.

An important property of the Wiener filter can be deduced if we analyze the gradient of the error surface at the optimal solution. The gradient vector can be expressed as follows:

$$\mathbf{g_w} = \frac{\partial E[e^2(k)]}{\partial \mathbf{w}} = E[2e(k)\frac{\partial e(k)}{\partial \mathbf{w}}] = -E[2e(k)\mathbf{x}(k)] \tag{2.94}$$

With the coefficients set at their optimal values, i.e., at the Wiener solution, the gradient vector is equal to zero, implying that

$$E[e(k)\mathbf{x}(k)] = \mathbf{0} \tag{2.95}$$

or

$$E[e(k)x(k-i)] = 0 \tag{2.96}$$

for $i = 0, 1, \ldots, N$. This means that the error signal is orthogonal to the elements of the input signal vector. In case either the error or the input signal has zero mean, the orthogonality property implies that $e(k)$ and $x(k)$ are uncorrelated.

The orthogonality principle also applies to the correlation between the output signal $y(k)$ and the error $e(k)$, when the tap weights are given by $\mathbf{w} = \mathbf{w}_o$. By premultiplying (2.95) by $\mathbf{w}_o^T$, the desired result follows:

$$E[e(k)\mathbf{w}_o^T\mathbf{x}(k)] = E[e(k)y(k)] = 0 \tag{2.97}$$

The gradient with respect to a complex parameter has not been defined. For our purposes the complex gradient vector can be defined as [18]

$$\mathbf{g}_{\mathbf{w}(k)}\{F(e(k))\} = \frac{1}{2}\left\{\frac{\partial F[e(k)]}{\partial \mathrm{re}[\mathbf{w}(k)]} - \jmath \frac{\partial F[e(k)]}{\partial \mathrm{im}[\mathbf{w}(k)]}\right\}$$

where re[·] and im[·] indicate real and imaginary parts of [·], respectively. Note that the partial derivatives are calculated for each element of $\mathbf{w}(k)$.

For the complex case the error signal and the MSE are, respectively, described by, see Chap. 14 for details,

$$e(k) = d(k) - \mathbf{w}^H(k)\mathbf{x}(k) \tag{2.98}$$

and

$$\begin{aligned}\xi &= E[|e(k)|^2] \\ &= E[|d(k)|^2] - 2\mathrm{re}\{\mathbf{w}^H E[d^*(k)\mathbf{x}(k)]\} + \mathbf{w}^H E[\mathbf{x}(k)\mathbf{x}^H(k)]\mathbf{w} \\ &= E[|d(k)|^2] - 2\mathrm{re}[\mathbf{w}^H\mathbf{p}] + \mathbf{w}^H\mathbf{R}\mathbf{w}\end{aligned} \tag{2.99}$$

where $\mathbf{p} = E[d^*(k)\mathbf{x}(k)]$ is the cross-correlation vector between the desired and input signals, and $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^H(k)]$ is the input signal correlation matrix. The Wiener solution in this case is also given by (2.92).

Example 2.2. The input signal of a first-order adaptive filter is described by

$$x(k) = \alpha_1 x_1(k) + \alpha_2 x_2(k)$$

where $x_1(k)$ and $x_2(k)$ are first-order AR processes and mutually uncorrelated having both unit variance. These signals are generated by applying distinct white noises to first-order filters whose poles are placed at $-s_1$ and $-s_2$, respectively.

(a) Calculate the autocorrelation matrix of the input signal.
(b) If the desired signal consists of $x_2(k)$, calculate the Wiener solution.

Solution. (a) The models for the signals involved are described by

$$x_i(k) = -s_i x_i(k-1) + \kappa_i n_i(k)$$

for $i = 1, 2$. According to (2.83) the autocorrelation of either $x_i(k)$ is given by

$$E[x_i(k)x_i(k-l)] = \kappa_i^2 \frac{(-s_i)^{|l|}}{1-s_i^2}\sigma_{n,i}^2 \tag{2.100}$$

where $\sigma_{n,i}^2$ is the variance of $n_i(k)$. Since each signal $x_i(k)$ has unit variance, then by applying $l = 0$ to the above equation

$$\kappa_i^2 = \frac{1-s_i^2}{\sigma_{n,i}^2} \tag{2.101}$$

Now by utilizing the fact that $x_1(k)$ and $x_2(k)$ are uncorrelated, the autocorrelation of the input signal is

$$\mathbf{R} = \begin{bmatrix} \alpha_1^2+\alpha_2^2 & -\alpha_1^2 s_1 - \alpha_2^2 s_2 \\ -\alpha_1^2 s_1 - \alpha_2^2 s_2 & \alpha_1^2+\alpha_2^2 \end{bmatrix}$$

$$\mathbf{p} = \begin{bmatrix} \alpha_2 \\ -\alpha_2 s_2 \end{bmatrix}$$

(b) The Wiener solution can then be expressed as

$$\begin{aligned}\mathbf{w}_o &= \mathbf{R}^{-1}\mathbf{p} \\ &= \frac{1}{(\alpha_1^2+\alpha_2^2)^2 - (\alpha_1^2 s_1 + \alpha_2^2 s_2)^2}\begin{bmatrix} \alpha_1^2+\alpha_2^2 & \alpha_1^2 s_1 + \alpha_2^2 s_2 \\ \alpha_1^2 s_1 + \alpha_2^2 s_2 & \alpha_1^2+\alpha_2^2 \end{bmatrix}\begin{bmatrix} \alpha_2 \\ -\alpha_2 s_2 \end{bmatrix}\end{aligned}$$

$$= \frac{1}{(1+\frac{\alpha_2^2}{\alpha_1^2})^2 - (s_1 + \frac{\alpha_2^2}{\alpha_1^2}s_2)^2} \begin{bmatrix} 1+\frac{\alpha_2^2}{\alpha_1^2} & s_1+\frac{\alpha_2^2}{\alpha_1^2}s_2 \\ s_1+\frac{\alpha_2^2}{\alpha_1^2}s_2 & 1+\frac{\alpha_2^2}{\alpha_1^2} \end{bmatrix} \begin{bmatrix} \frac{\alpha_2}{\alpha_1^2} \\ -\frac{\alpha_2}{\alpha_1^2}s_2 \end{bmatrix}$$

$$= \alpha_2 \begin{bmatrix} \frac{1}{\alpha_1^2+\alpha_2^2-s_1\alpha_1^2-s_2\alpha_2^2} & 0 \\ 0 & \frac{1}{\alpha_1^2+\alpha_2^2+s_1\alpha_1^2+s_2\alpha_2^2} \end{bmatrix} \begin{bmatrix} \frac{1-s_2}{2} \\ -\frac{1+s_2}{2} \end{bmatrix}$$

Let's assume that in this example our task was to detect the presence of $x_2(k)$ in the input signal. For a fixed input-signal power, from this solution it is possible to observe that lower signal to interference at the input, that is lower $\frac{\alpha_2^2}{\alpha_1^2}$, leads to a Wiener solution vector with lower norm. This result reflects the fact that the Wiener solution tries to detect the desired signal at the same time it avoids enhancing the undesired signal, i.e., the interference $x_1(k)$. □

2.5 Linearly Constrained Wiener Filter

In a number of applications, it is required to impose some linear constraints on the filter coefficients such that the optimal solution is the one that achieves the minimum MSE, provided the constraints are met. Typical constraints are: unity norm of the parameter vector; linear phase of the adaptive filter; prescribed gains at given frequencies.

In the particular case of an array of antennas the measured signals can be linearly combined to form a directional beam, where the signal impinging on the array in the desired direction will have higher gain. This application is called beamforming, where we specify gains at certain directions of arrival. It is clear that the array is introducing another dimension to the received data, namely spatial information. The weights in the antennas can be made adaptive leading to the so-called adaptive antenna arrays. This is the principle behind the concept of smart antennas, where a set of adaptive array processors filter the signals coming from the array, and direct the beam to several different directions where a potential communication is required. For example, in a wireless communication system we are able to form a beam for each subscriber according to its position, ultimately leading to minimization of noise from the environment and interference from other subscribers.

In order to develop the theory of linearly constrained optimal filters, let us consider the particular application of a narrowband beamformer required to pass without distortion all signals arriving at 90° with respect to the array of antennas. All other sources of signals shall be treated as interferers and must be attenuated as much as possible. Figure 2.2 illustrates the application. Note that in case the signal of interest does not impinge the array at 90° with respect to the array, a steering operation in the constraint vector **c** (to be defined) has to be performed [23].

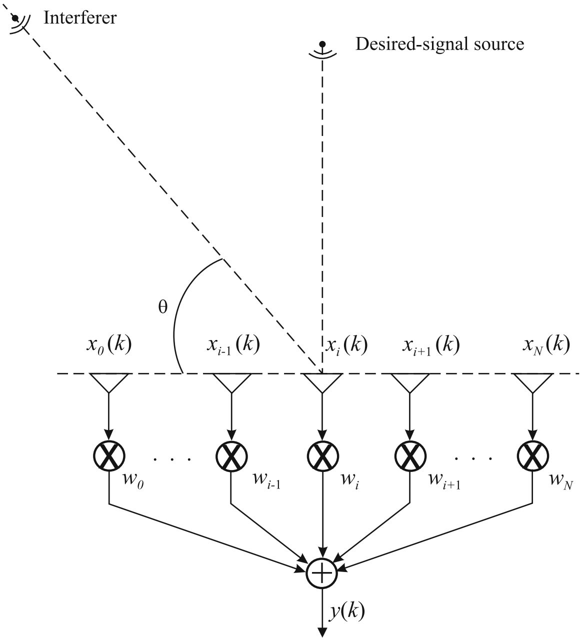

Fig. 2.2 Narrowband beamformer

The optimal filter that satisfies the linear constraints is called the *linearly constrained minimum-variance* (LCMV) filter.

If the desired signal source is sufficiently far from the array of antennas, then we may assume that the wavefronts are planar at the array. Therefore, the wavefront from the desired source will reach all antennas at the same instant, whereas the wavefront from the interferer will reach each antenna at different time instants. Taking the antenna with input signal x_0 as a time reference t_0, the wavefront will reach the ith antenna at [23]

$$t_i = t_0 + i\,\frac{d\cos\theta}{c}$$

where θ is the angle between the antenna array and the interferer direction of arrival, d is the distance between neighboring antennas, and c is the speed of propagation of the wave (3×10^8 m/s).

For this particular case, the LCMV filter is the one that minimizes the array output signal energy

$$\begin{aligned} \xi = E[y^2(k)] &= E[\mathbf{w}^T\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}] \\ \text{subject to :} \quad & \sum_{j=0}^{N} c_j w_j = f \end{aligned} \tag{2.102}$$

where

$$\mathbf{w} = [w_0 \, w_1 \ldots w_N]^T$$
$$\mathbf{x}(k) = [x_0(k) \, x_1(k) \ldots x_N(k)]^T$$

and

$$\mathbf{c} = [1 \, 1 \ldots 1]^T$$

is the constraint vector, since $\theta = 90°$. The desired gain is usually $f = 1$.

In the case the desired signal impinges the array at an angle θ with respect to the array, the incoming signal reaches the ith antenna delayed by $i\,\frac{d\cos\theta}{c}$ with respect to the 0th antenna [24]. Let's consider the case of a narrowband array such that all antennas detect the impinging signal with the same amplitude when measured taking into consideration their relative delays, which are multiples of $\frac{d\cos\theta}{c}$. In such a case the optimal receiver coefficients would be

$$w_i = \frac{\mathrm{e}^{\jmath\omega\tau_i}}{N+1} \tag{2.103}$$

for $i = 0, 1, \ldots, N$, in order to add coherently the delays of the desired incoming signal at a given direction θ. The impinging signal appears at the ith antenna multiplied by $\mathrm{e}^{-\jmath\omega\tau_i}$, considering the particular case of array configuration of Fig. 2.2. In this uniform linear array, the antenna locations are

$$p_i = id$$

for $i = 0, 1, \ldots, N$. Using the 0th antenna as reference, the signal will reach the array according to the following pattern

$$\begin{aligned}\tilde{\mathbf{c}} &= \mathrm{e}^{\jmath\omega t}\left[1 \; \mathrm{e}^{-\jmath\omega\frac{d\cos\theta}{c}} \; \mathrm{e}^{-\jmath\omega\frac{2d\cos\theta}{c}} \ldots \mathrm{e}^{-\jmath\omega\frac{Nd\cos\theta}{c}}\right]^T \\ &= \mathrm{e}^{\jmath\omega t}\left[1 \; \mathrm{e}^{-\jmath\frac{2\pi}{\lambda}d\cos\theta} \; \mathrm{e}^{-\jmath\frac{2\pi}{\lambda}2d\cos\theta} \ldots \mathrm{e}^{-\jmath\frac{2\pi}{\lambda}Nd\cos\theta}\right]^T\end{aligned} \tag{2.104}$$

where the equality $\frac{\omega}{c} = \frac{2\pi}{\lambda}$ was employed, with λ being the wavelength corresponding to the frequency ω.

By defining the variable $\psi(\omega,\theta) = \frac{2\pi}{\lambda}d\cos\theta$, we can describe the output signal of the beamformer as

$$\begin{aligned}y &= \mathrm{e}^{\jmath\omega t}\sum_{i=0}^{N} w_i \mathrm{e}^{-\jmath\psi(\omega,\theta)i} \\ &= \mathrm{e}^{\jmath\omega t} H(\omega,\theta)\end{aligned} \tag{2.105}$$

where $H(\omega, \theta)$ modifies the amplitude and phase of transmitted signal at a given frequency ω. Note that the shaping function $H(\omega, \theta)$ depends on the impinging angle.

For the sake of illustration, if the antenna separation is $d = \frac{\lambda}{2}$, $\theta = 60°$, and N is odd, then the constraint vector would be

$$\begin{aligned}\mathbf{c} &= \left[1 \;\; \mathrm{e}^{-\jmath\frac{\pi}{2}} \;\; \mathrm{e}^{-\jmath\pi} \ldots \mathrm{e}^{-\jmath\frac{N\pi}{2}}\right]^T \\ &= \left[1 \;\; -\jmath \;\; -1 \ldots \mathrm{e}^{-\jmath\frac{N\pi}{2}}\right]^T \end{aligned} \tag{2.106}$$

Using the method of Lagrange multipliers, we can rewrite the constrained minimization problem described in (2.102) as

$$\xi_{\mathrm{c}} = E[\mathbf{w}^T \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}] + \lambda(\mathbf{c}^T\mathbf{w} - f) \tag{2.107}$$

The gradient of ξ_{c} with respect to $\mathbf{w}$ is equal to

$$\mathbf{g_w} = 2\mathbf{R}\mathbf{w} + \lambda\mathbf{c} \tag{2.108}$$

where $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$. For a positive definite matrix $\mathbf{R}$, the value of $\mathbf{w}$ that satisfies $\mathbf{g_w} = \mathbf{0}$ is unique and minimizes ξ_c. Denoting $\mathbf{w}_o$ as the optimal solution, we have

$$\begin{aligned} &2\mathbf{R}\mathbf{w}_o + \lambda\mathbf{c} = \mathbf{0} \\ &2\mathbf{c}^T\mathbf{w}_o + \lambda\mathbf{c}^T\mathbf{R}^{-1}\mathbf{c} = \mathbf{0} \\ &2f + \lambda\mathbf{c}^T\mathbf{R}^{-1}\mathbf{c} = \mathbf{0} \end{aligned}$$

where in order to obtain the second equality, we premultiply the first equation by $\mathbf{c}^T\mathbf{R}^{-1}$. Therefore,

$$\lambda = -2(\mathbf{c}^T\mathbf{R}^{-1}\mathbf{c})^{-1} f$$

and the LCMV filter is

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{c}(\mathbf{c}^T\mathbf{R}^{-1}\mathbf{c})^{-1} f \tag{2.109}$$

If more constraints need to be satisfied by the filter, these can be easily incorporated in a constraint matrix and in a gain vector, such that

$$\mathbf{C}^T\mathbf{w} = \mathbf{f} \tag{2.110}$$

In this case, the LCMV filter is given by

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C})^{-1}\mathbf{f} \tag{2.111}$$

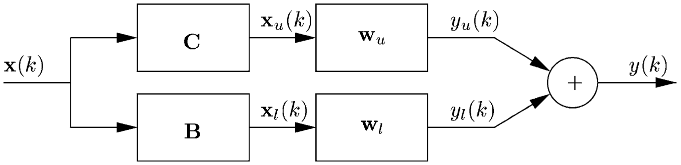

Fig. 2.3 The generalized sidelobe canceller

If there is a desired signal, the natural objective is the minimization of the MSE, not the output energy as in the narrowband beamformer. In this case, it is straightforward to modify (2.107) and obtain the optimal solution

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} + \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C})^{-1}(\mathbf{f} - \mathbf{C}^T\mathbf{R}^{-1}\mathbf{p}) \tag{2.112}$$

where $\mathbf{p} = E[d(k)\,\mathbf{x}(k)]$, see Problem 20.

In the case of complex input signals and constraints, the optimal solution is given by

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} + \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^H\mathbf{R}^{-1}\mathbf{C})^{-1}(\mathbf{f} - \mathbf{C}^H\mathbf{R}^{-1}\mathbf{p}) \tag{2.113}$$

where $\mathbf{C}^H\mathbf{w} = \mathbf{f}$.

2.5.1 The Generalized Sidelobe Canceller

An alternative implementation to the direct-form constrained adaptive filter showed above is called the generalized sidelobe canceller (GSC) (see Fig. 2.3) [25].

For this structure the input signal vector is transformed by a matrix

$$\mathbf{T} = [\mathbf{C}\;\mathbf{B}] \tag{2.114}$$

where $\mathbf{C}$ is the constraint matrix and $\mathbf{B}$ is a *blocking matrix* that spans the null space of $\mathbf{C}$, i.e., matrix $\mathbf{B}$ satisfies

$$\mathbf{B}^T\mathbf{C} = \mathbf{0} \tag{2.115}$$

The output signal $y(k)$ shown in Fig. 2.3 is formed as

$$\begin{aligned} y(k) &= \mathbf{w}_u^T\mathbf{C}^T\mathbf{x}(k) + \mathbf{w}_l^T\mathbf{B}^T\mathbf{x}(k) \\ &= (\mathbf{C}\mathbf{w}_u + \mathbf{B}\mathbf{w}_l)^T\mathbf{x}(k) \\ &= (\mathbf{T}\mathbf{w})^T\mathbf{x}(k) \\ &= \bar{\mathbf{w}}^T\mathbf{x}(k) \end{aligned} \tag{2.116}$$

where $\mathbf{w} = [\mathbf{w}_u^T\;\mathbf{w}_l^T]^T$ and $\bar{\mathbf{w}} = \mathbf{T}\mathbf{w}$.

The linear constraints are satisfied if $\mathbf{C}^T\bar{\mathbf{w}} = \mathbf{f}$. But as $\mathbf{C}^T\mathbf{B} = \mathbf{0}$, then the condition to be satisfied becomes

$$\mathbf{C}^T\bar{\mathbf{w}} = \mathbf{C}^T\mathbf{C}\mathbf{w}_u = \mathbf{f} \tag{2.117}$$

Therefore, for the GSC structure shown in Fig. 2.3 there is a necessary condition that the upper part of the coefficient vector, $\mathbf{w}_u$, should be initialized as

$$\mathbf{w}_u = (\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f} \tag{2.118}$$

Minimization of the output energy is achieved with a proper choice of $\mathbf{w}_l$. In fact, we transformed a constrained optimization problem into an unconstrained one, which in turn can be solved with the classical linear Wiener filter, i.e.,

$$\begin{aligned}\min_{\mathbf{w}_l} E[y^2(k)] &= \min_{\mathbf{w}_l} E\{[y_u(k) + \mathbf{w}_l^T\mathbf{x}_l(k)]^2\}\\ &= \mathbf{w}_{l,o}\\ &= -\mathbf{R}_l^{-1}\mathbf{p}_l,\end{aligned} \tag{2.119}$$

where

$$\begin{aligned}\mathbf{R}_l &= E[\mathbf{x}_l(k)\mathbf{x}_l^T(k)]\\ &= E[\mathbf{B}^T\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{B}]\\ &= \mathbf{B}^T[\mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{B}\\ &= \mathbf{B}^T\mathbf{R}\mathbf{B}\end{aligned} \tag{2.120}$$

and

$$\begin{aligned}\mathbf{p}_l &= E[y_u(k)\,\mathbf{x}_l(k)] = E[\mathbf{x}_l(k)\,y_u(k)]\\ &= E[\mathbf{B}^T\mathbf{x}(k)\,\mathbf{w}_u^T\mathbf{C}^T\mathbf{x}(k)]\\ &= E[\mathbf{B}^T\mathbf{x}(k)\,\mathbf{x}^T(k)\mathbf{C}\mathbf{w}_u]\\ &= \mathbf{B}^T E[\mathbf{x}(k)\,\mathbf{x}^T(k)]\mathbf{C}\mathbf{w}_u\\ &= \mathbf{B}^T\mathbf{R}\mathbf{C}\mathbf{w}_u\\ &= \mathbf{B}^T\mathbf{R}\mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f}\end{aligned} \tag{2.121}$$

where in the above derivations we utilized the results and definitions from (2.116) and (2.118).

Using (2.118), (2.120), and (2.121) it is possible to show that

$$\mathbf{w}_{l,o} = -(\mathbf{B}^T\mathbf{R}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{R}\mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f} \tag{2.122}$$

Given that $\mathbf{w}_{l,o}$ is the solution to an unconstrained minimization problem of transformed quantities, any unconstrained adaptive filter can be used to estimate recursively this optimal solution. The drawback in the implementation of the GSC structure comes from the transformation of the input signal vector via a constraint matrix and a blocking matrix. Although in theory any matrix with linearly independent columns that spans the null space of $\mathbf{C}$ can be employed, in many cases the computational complexity resulting from the multiplication of $\mathbf{B}$ by $\mathbf{x}(k)$ can be prohibitive. Furthermore, if the transformation matrix $\mathbf{T}$ is not orthogonal, finite-precision effects may yield an overall unstable system. A simple solution that guarantees orthogonality in the transformation and low computational complexity can be obtained with a Householder transformation [26].

2.6 MSE Surface

The MSE is a quadratic function of the parameters $\mathbf{w}$. Assuming a given fixed $\mathbf{w}$, the MSE is not a function of time and can be expressed as

$$\xi = \sigma_d^2 - 2\mathbf{w}^T\mathbf{p} + \mathbf{w}^T\mathbf{R}\mathbf{w} \tag{2.123}$$

where σ_d^2 is the variance of $d(k)$ assuming it has zero-mean. The MSE is a quadratic function of the tap weights forming a hyperparaboloid surface. The MSE surface is convex and has only positive values. For two weights, the surface is a paraboloid. Figure 2.4 illustrates the MSE surface for a numerical example where $\mathbf{w}$ has two coefficients. If the MSE surface is intersected by a plane parallel to the $\mathbf{w}$ plane, placed at a level superior to $\xi_{\min}$, the intersection consists of an ellipse representing equal MSE contours as depicted in Fig. 2.5. Note that in this figure we showed three distinct ellipses, corresponding to different levels of MSE. The ellipses of constant MSE are all concentric. In order to understand the properties of the MSE surface, it is convenient to define a translated coefficient vector as follows:

$$\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}_o \tag{2.124}$$

The MSE can be expressed as a function of $\Delta\mathbf{w}$ as follows:

$$\begin{aligned}
\xi &= \sigma_d^2 - \mathbf{w}_o^T\mathbf{p} + \mathbf{w}_o^T\mathbf{p} - 2\mathbf{w}^T\mathbf{p} + \mathbf{w}^T\mathbf{R}\mathbf{w} \\
&= \xi_{\min} - \Delta\mathbf{w}^T\mathbf{p} - \mathbf{w}^T\mathbf{R}\mathbf{w}_o + \mathbf{w}^T\mathbf{R}\mathbf{w} \\
&= \xi_{\min} - \Delta\mathbf{w}^T\mathbf{p} + \mathbf{w}^T\mathbf{R}\Delta\mathbf{w} \\
&= \xi_{\min} - \mathbf{w}_o^T\mathbf{R}\Delta\mathbf{w} + \mathbf{w}^T\mathbf{R}\Delta\mathbf{w} \\
&= \xi_{\min} + \Delta\mathbf{w}^T\mathbf{R}\Delta\mathbf{w}
\end{aligned} \tag{2.125}$$

where we used the results of (2.92) and (2.93). The corresponding error surface contours are depicted in Fig. 2.6.

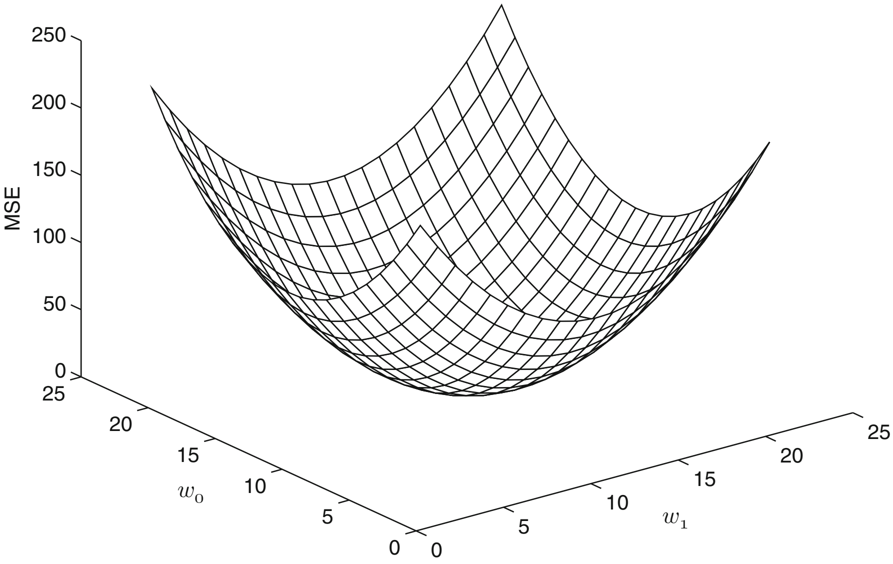

Fig. 2.4 Mean-square error surface

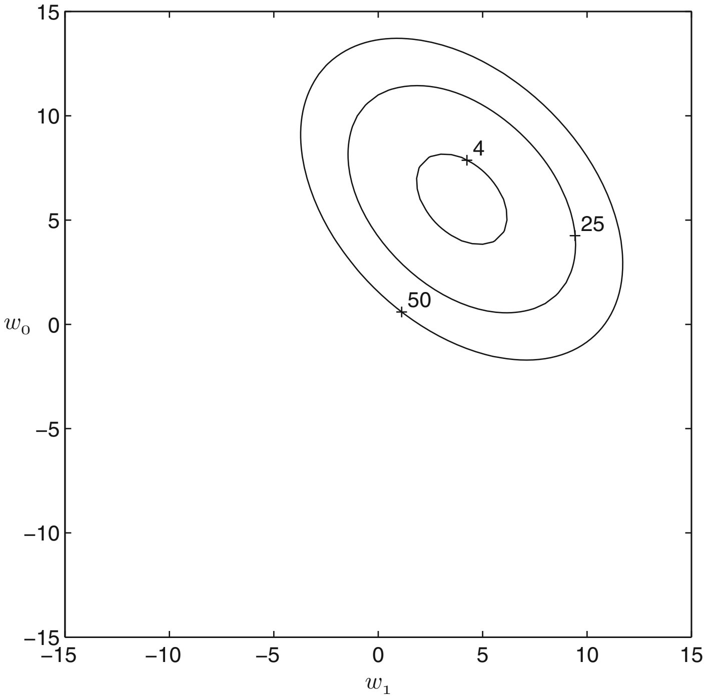

Fig. 2.5 Contours of the MSE surface

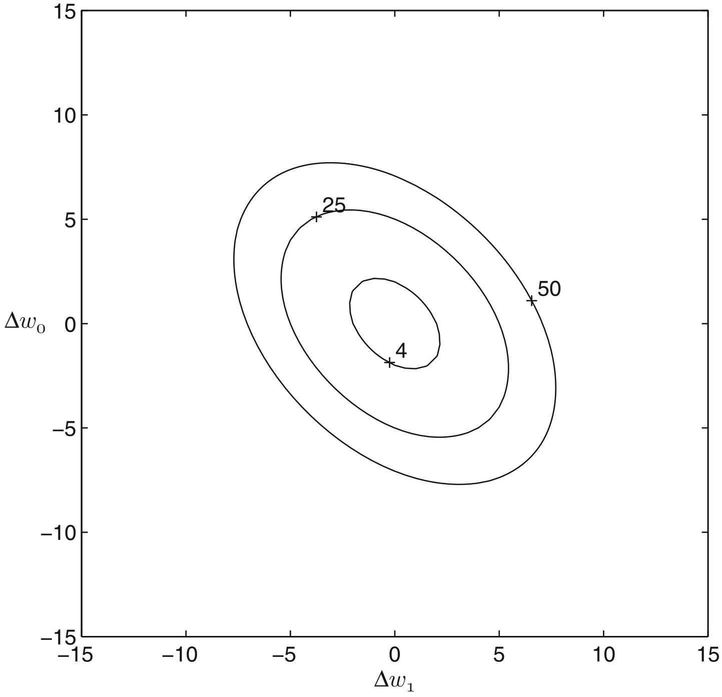

Fig. 2.6 Translated contours of the MSE surface

By employing the diagonalized form of **R**, the last equation can be rewritten as follows:

$$\begin{aligned}\xi &= \xi_{\min} + \Delta\mathbf{w}^T\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^T\Delta\mathbf{w}\\ &= \xi_{\min} + \mathbf{v}^T\boldsymbol{\Lambda}\mathbf{v}\\ &= \xi_{\min} + \sum_{i=0}^{N}\lambda_i v_i^2 \end{aligned} \tag{2.126}$$

where $\mathbf{v} = \mathbf{Q}^T\Delta\mathbf{w}$ are the rotated parameters.

The above form for representing the MSE surface is an uncoupled form, in the sense that each component of the gradient vector of the MSE with respect to the rotated parameters is a function of a single parameter, that is

$$\mathbf{g_v}[\xi] = [2\lambda_0 v_0 \;\; 2\lambda_1 v_1 \;\ldots\; 2\lambda_N v_N]^T$$

This property means that if all v_i's are zero except one, the gradient direction coincides with the nonzero parameter axis. In other words, the rotated parameters

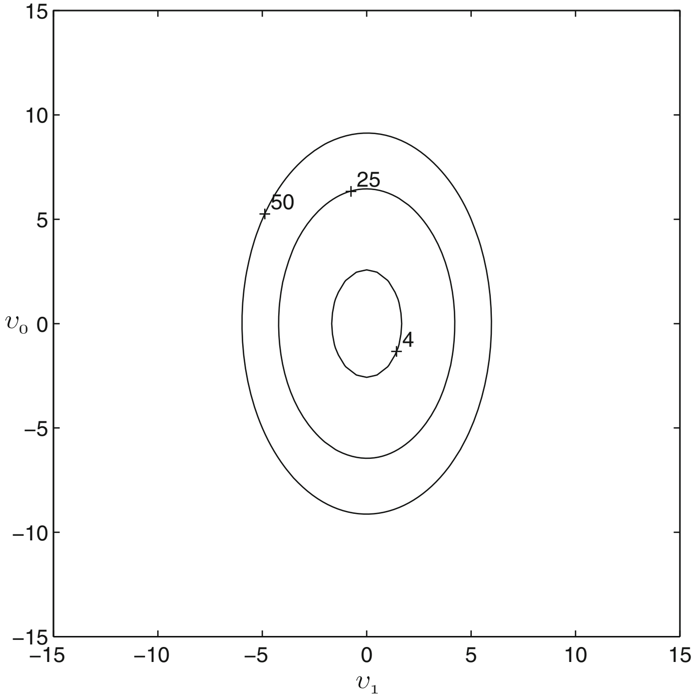

Fig. 2.7 Rotated contours of the MSE surface

represent the principal axes of the hyperellipse of constant MSE, as illustrated in Fig. 2.7. Note that since the rotated parameters are the result of the projection of the original parameter vector $\Delta\mathbf{w}$ on the eigenvectors $\mathbf{q}_i$ direction, it is straightforward to conclude that the eigenvectors represent the principal axes of the constant MSE hyperellipses.

The matrix of second derivatives of ξ as related to the rotated parameters is $\boldsymbol{\Lambda}$. We can note that the gradient will be steeper in the principal axes corresponding to larger eigenvalues. This is the direction, in the two axes case, where the ellipse is narrow.

2.7 Bias and Consistency

The correct interpretation of the results obtained by the adaptive-filtering algorithm requires the definitions of bias and consistency. An estimate is considered unbiased if the following condition is satisfied

$$E[\mathbf{w}(k)] = \mathbf{w}_o \tag{2.127}$$

The difference $E[\mathbf{w}(k)] - \mathbf{w}_o$ is called the bias in the parameter estimate.

An estimate is considered consistent if

$$\mathbf{w}(k) \to \mathbf{w}_o \text{ as } k \to \infty \tag{2.128}$$

Note that since $\mathbf{w}(k)$ is a random variable, it is necessary to define in which sense the limit is taken. Usually, the limit with probability one is employed. In the case of identification, a system is considered identifiable if the given parameter estimates are consistent. For a more formal treatment on this subject, refer to [21].

2.8 Newton Algorithm

In the context of the MSE minimization discussed in the previous section, see (2.123), the coefficient-vector updating using the Newton method is performed as follows:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{R}^{-1}\mathbf{g}_{\mathbf{w}}(k) \tag{2.129}$$

where its derivation originates from (1.4). Assuming the true gradient and the matrix $\mathbf{R}$ are available, the coefficient-vector updating can be expressed as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{R}^{-1}[-2\mathbf{p} + 2\mathbf{R}\mathbf{w}(k)] = (\mathbf{I} - 2\mu\mathbf{I})\mathbf{w}(k) + 2\mu\mathbf{w}_o \tag{2.130}$$

where if $\mu = 1/2$, the Wiener solution is reached in one step.

The Wiener solution can be approached using a Newton-like search algorithm, by updating the adaptive-filter coefficients as follows:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \hat{\mathbf{R}}^{-1}(k)\hat{\mathbf{g}}_{\mathbf{w}}(k) \tag{2.131}$$

where $\hat{\mathbf{R}}^{-1}(k)$ is an estimate of $\mathbf{R}^{-1}$ and $\hat{\mathbf{g}}_{\mathbf{w}}(k)$ is an estimate of $\mathbf{g}_{\mathbf{w}}$, both at instant k. The parameter μ is the convergence factor that regulates the convergence rate. Newton-based algorithms present, in general, fast convergence. However, the estimate of $\mathbf{R}^{-1}$ is computationally intensive and can become numerically unstable if special care is not taken. These factors made the steepest-descent-based algorithms more popular in adaptive-filtering applications.

2.9 Steepest-Descent Algorithm

In order to get a practical feeling of a problem that is being solved using the steepest-descent algorithm, we assume that the optimal coefficient vector, i.e., the Wiener solution, is $\mathbf{w}_o$, and that the reference signal is not corrupted by measurement noise.[8]

[8] Noise added to the reference signal originated from environment and/or thermal noise.

The main objective of the present section is to study the rate of convergence, the stability, and the steady-state behavior of an adaptive filter whose coefficients are updated through the steepest-descent algorithm. It is worth mentioning that the steepest-descent method can be considered an efficient gradient-type algorithm, in the sense that it works with the true gradient vector, and not with an estimate of it. Therefore, the performance of other gradient-type algorithms can at most be close to the performance of the steepest-descent algorithm. When the objective function is the MSE, the difficult task of obtaining the matrix **R** and the vector **p** impairs the steepest-descent algorithm from being useful in adaptive-filtering applications. Its performance, however, serves as a benchmark for gradient-based algorithms.

The steepest-descent algorithm updates the coefficients in the following general form

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{g}_{\mathbf{w}}(k) \tag{2.132}$$

where the above expression is equivalent to (1.6). It is worth noting that several alternative gradient-based algorithms available replace $\mathbf{g}_{\mathbf{w}}(k)$ by an estimate $\hat{\mathbf{g}}_{\mathbf{w}}(k)$, and they differ in the way the gradient vector is estimated. The true gradient expression is given in (2.91) and, as can be noted, it depends on the vector **p** and the matrix **R**, that are usually not available.

Substituting (2.91) in (2.132), we get

$$\mathbf{w}(k+1) = \mathbf{w}(k) - 2\mu \mathbf{R}\mathbf{w}(k) + 2\mu \mathbf{p} \tag{2.133}$$

Now, some of the main properties related to the convergence behavior of the steepest-descent algorithm in stationary environment are described. First, an analysis is required to determine the influence of the convergence factor μ in the convergence behavior of the steepest-descent algorithm.

The error in the adaptive-filter coefficients when compared to the Wiener solution is defined as

$$\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o \tag{2.134}$$

The steepest-descent algorithm can then be described in an alternative way, that is:

$$\begin{aligned}\Delta\mathbf{w}(k+1) &= \Delta\mathbf{w}(k) - 2\mu[\mathbf{R}\mathbf{w}(k) - \mathbf{R}\mathbf{w}_o] \\ &= \Delta\mathbf{w}(k) - 2\mu\mathbf{R}\Delta\mathbf{w}(k) \\ &= (\mathbf{I} - 2\mu\mathbf{R})\,\Delta\mathbf{w}(k)\end{aligned} \tag{2.135}$$

where the relation $\mathbf{p} = \mathbf{R}\mathbf{w}_o$ (see (2.92)) was employed. It can be shown from the above equation that

$$\Delta\mathbf{w}(k+1) = (\mathbf{I} - 2\mu\mathbf{R})^{k+1}\Delta\mathbf{w}(0) \tag{2.136}$$

or

$$\mathbf{w}(k+1) = \mathbf{w}_o + (\mathbf{I} - 2\mu\mathbf{R})^{k+1}[\mathbf{w}(0) - \mathbf{w}_o] \tag{2.137}$$

The (2.135) premultiplied by $\mathbf{Q}^T$, where $\mathbf{Q}$ is the unitary matrix that diagonalizes $\mathbf{R}$ through a similarity transformation, yields

$$\begin{aligned}\mathbf{Q}^T\Delta\mathbf{w}(k+1) &= (\mathbf{I}-2\mu\mathbf{Q}^T\mathbf{R}\mathbf{Q})\mathbf{Q}^T\Delta\mathbf{w}(k)\\ &= \mathbf{v}(k+1)\\ &= (\mathbf{I}-2\mu\boldsymbol{\Lambda})\mathbf{v}(k)\\ &= \begin{bmatrix} 1-2\mu\lambda_0 & 0 & \cdots & 0\\ 0 & 1-2\mu\lambda_1 & & \vdots\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & & 1-2\mu\lambda_N\end{bmatrix}\mathbf{v}(k)\end{aligned} \quad (2.138)$$

In the above equation, $\mathbf{v}(k+1) = \mathbf{Q}^T\Delta\mathbf{w}(k+1)$ is the rotated coefficient-vector error. Using induction, (2.138) can be rewritten as

$$\begin{aligned}\mathbf{v}(k+1) &= (\mathbf{I}-2\mu\boldsymbol{\Lambda})^{k+1}\mathbf{v}(0)\\ &= \begin{bmatrix} (1-2\mu\lambda_0)^{k+1} & 0 & \cdots & 0\\ 0 & (1-2\mu\lambda_1)^{k+1} & & \vdots\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & & (1-2\mu\lambda_N)^{k+1}\end{bmatrix}\mathbf{v}(0)\end{aligned} \quad (2.139)$$

This equation shows that in order to guarantee the convergence of the coefficients, each element $1-2\mu\lambda_i$ must have an absolute value less than one. As a consequence, the convergence factor of the steepest-descent algorithm must be chosen in the range

$$0 < \mu < \frac{1}{\lambda_{\max}} \quad (2.140)$$

where $\lambda_{\max}$ is the largest eigenvalue of $\mathbf{R}$. In this case, all the elements of the diagonal matrix in (2.139) tend to zero as $k \to \infty$, resulting in $\mathbf{v}(k+1) \to 0$ for large k.

The μ value in the above range guarantees that the coefficient vector approaches the optimum coefficient vector $\mathbf{w}_o$. It should be mentioned that if matrix $\mathbf{R}$ has large eigenvalue spread, the convergence speed of the coefficients will be primarily dependent on the value of the smallest eigenvalue. Note that the slowest decaying element in (2.139) is given by $(1-2\mu\lambda_{\min})^{k+1}$.

The MSE presents a transient behavior during the adaptation process that can be analyzed in a straightforward way if we employ the diagonalized version of $\mathbf{R}$. Recalling from (2.125) that

$$\xi(k) = \xi_{\min} + \Delta\mathbf{w}^T(k)\mathbf{R}\Delta\mathbf{w}(k) \quad (2.141)$$

the MSE can then be simplified as follows:

$$\begin{aligned}\xi(k) &= \xi_{\min} + \Delta\mathbf{w}^T(k)\mathbf{Q}\boldsymbol{\Lambda}\,\mathbf{Q}^T\Delta\mathbf{w}(k)\\ &= \xi_{\min} + \mathbf{v}^T(k)\boldsymbol{\Lambda}\,\mathbf{v}(k)\\ &= \xi_{\min} + \sum_{i=0}^{N}\lambda_i v_i^2(k) \qquad (2.142)\end{aligned}$$

If we apply the result of (2.139) in (2.142), it can be shown that the following relation results

$$\begin{aligned}\xi(k) &= \xi_{\min} + \mathbf{v}^T(k-1)(\mathbf{I}-2\mu\boldsymbol{\Lambda})\boldsymbol{\Lambda}\,(\mathbf{I}-2\mu\boldsymbol{\Lambda})\mathbf{v}(k-1)\\ &= \xi_{\min} + \sum_{i=0}^{N}\lambda_i(1-2\mu\lambda_i)^{2k}v_i^2(0) \qquad (2.143)\end{aligned}$$

The analyses presented in this section show that before the steepest-descent algorithm reaches the steady-state behavior, there is a transient period where the error is usually high and the coefficients are far from the Wiener solution. As can be seen from (2.139), in the case of the adaptive-filter coefficients, the convergence will follow $(N+1)$ geometric decaying curves with ratios $r_{wi} = (1-2\mu\lambda_i)$. Each of these curves can be approximated by an exponential envelope with time constant τ_{wi} as follows [5]:

$$r_{wi} = \mathrm{e}^{\frac{-1}{\tau_{wi}}} = 1 - \frac{1}{\tau_{wi}} + \frac{1}{2!\tau_{wi}^2} + \cdots \qquad (2.144)$$

In general, r_{wi} is slightly smaller than one, specially in the cases of slowly decreasing modes that correspond to small values λ_i and μ. Therefore,

$$r_{wi} = (1-2\mu\lambda_i) \approx 1 - \frac{1}{\tau_{wi}} \qquad (2.145)$$

then

$$\tau_{wi} \approx \frac{1}{2\mu\lambda_i}$$

for $i = 0, 1, \ldots, N$.

For the convergence of the MSE, the range of values of μ is the same to guarantee the convergence of the coefficients. In this case, due to the exponent $2k$ in (2.143), the geometric decaying curves have ratios given by $r_{ei} = (1-4\mu\lambda_i)$, that can be approximated by exponential envelopes with time constants given by

$$\tau_{ei} \approx \frac{1}{4\mu\lambda_i} \qquad (2.146)$$

for $i = 0, 1, \ldots, N$, where it was considered that $4\mu^2\lambda_i^2 \ll 1$. In the convergence of both the error and the coefficients, the time required for the convergence depends on the ratio of the eigenvalues of the input signal. Further discussions on convergence properties that apply to gradient-type algorithms can be found in Chap. 3.

Example 2.3. The matrix **R** and the vector **p** are known for a given experimental environment:

$$\mathbf{R} = \begin{bmatrix} 1 & 0.4045 \\ 0.4045 & 1 \end{bmatrix}$$

$$\mathbf{p} = [0 \;\; 0.2939]^T$$

$$E[d^2(k)] = 0.5$$

(a) Deduce the equation for the MSE.
(b) Choose a small value for μ, and starting the parameters at $[-1 \quad -2]^T$ plot the convergence path of the steepest-descent algorithm in the MSE surface.
(c) Repeat the previous item for the Newton algorithm starting at $[0 \;\; -2]^T$.

Solution. (a) The MSE function is given by

$$\begin{aligned}\xi &= E[d^2(k)] - 2\mathbf{w}^T\mathbf{p} + \mathbf{w}^T\mathbf{R}\mathbf{w} \\ &= \sigma_d^2 - 2[w_1 \;\; w_2]\begin{bmatrix} 0 \\ 0.2939 \end{bmatrix} + [w_1 \;\; w_2]\begin{bmatrix} 1 & 0.4045 \\ 0.4045 & 1 \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}\end{aligned}$$

After performing the algebraic calculations, we obtain the following result

$$\xi = 0.5 + w_1^2 + w_2^2 + 0.8090 w_1 w_2 - 0.5878 w_2$$

(b) The steepest-descent algorithm was applied to minimize the MSE using a convergence factor $\mu = 0.1/\lambda_{\max}$, where $\lambda_{\max} = 1.4045$. The convergence path of the algorithm in the MSE surface is depicted in Fig. 2.8. As can be noted, the path followed by the algorithm first approaches the main axis (eigenvector) corresponding to the smaller eigenvalue, and then follows toward the minimum in a direction increasingly aligned with this main axis.
(c) The Newton algorithm was also applied to minimize the MSE using a convergence factor $\mu = 0.1/\lambda_{\max}$. The convergence path of the Newton algorithm in the MSE surface is depicted in Fig. 2.9. The Newton algorithm follows a straight path to the minimum. □

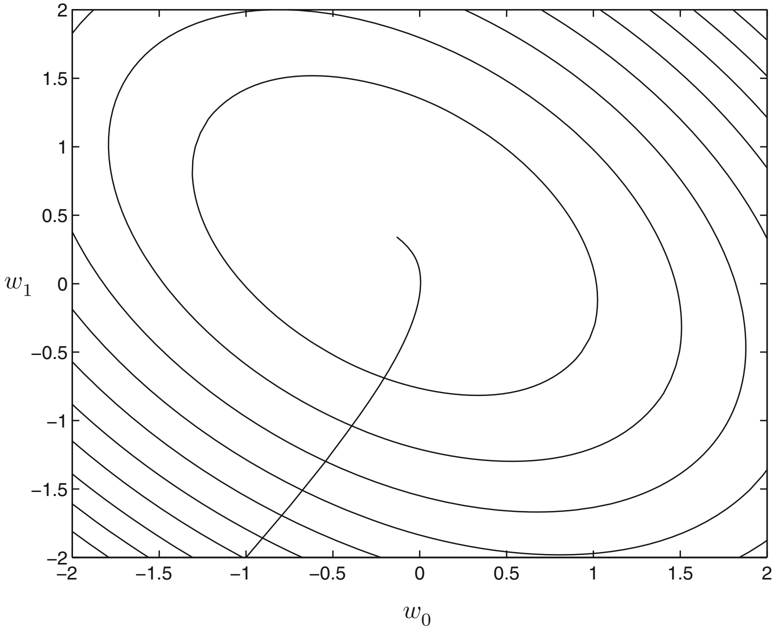

Fig. 2.8 Convergence path of the steepest-descent algorithm

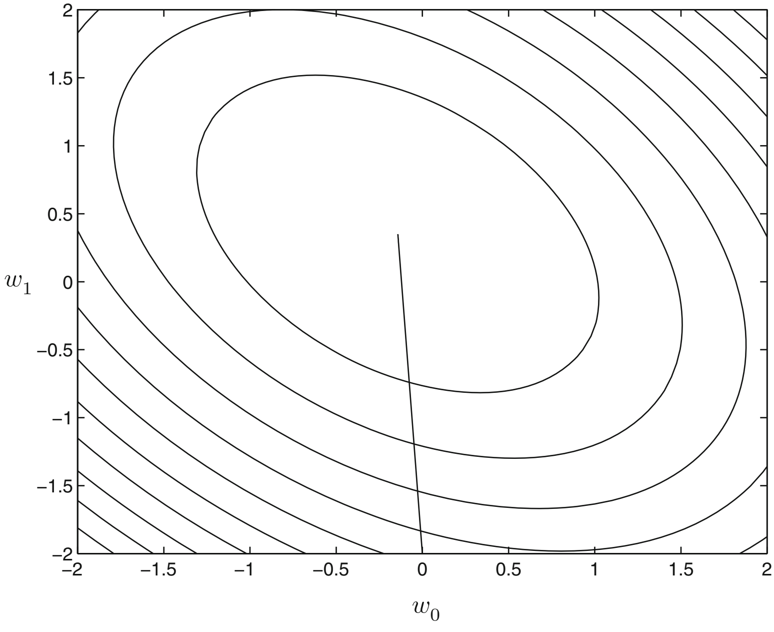

Fig. 2.9 Convergence path of the Newton algorithm

2.10 Applications Revisited

In this section, we give a brief introduction to the typical applications where the adaptive-filtering algorithms are required, including a discussion of where in the real world these applications are found. The main objective of this section is to illustrate how the adaptive-filtering algorithms, in general, and the ones presented in the book, in particular, are applied to solve practical problems. It should be noted that the detailed analysis of any particular application is beyond the scope of this book. Nevertheless, a number of specific references are given for the interested reader. The distinctive feature of each application is the way the adaptive filter input signal and the desired signal are chosen. Once these signals are determined, any known properties of them can be used to understand the expected behavior of the adaptive filter when attempting to minimize the chosen objective function (for example, the MSE, ξ).

2.10.1 System Identification

The typical setup of the system identification application is depicted in Fig. 2.10. A common input signal is applied to the unknown system and to the adaptive filter. Usually, the input signal is a wideband signal, in order to allow the adaptive filter to converge to a good model of the unknown system.

Assume the unknown system has impulse response given by $h(k)$, for $k = 0, 1, 2, 3, \ldots, \infty$, and zero for $k < 0$. The error signal is then given by

$$\begin{aligned} e(k) &= d(k) - y(k) \\ &= \sum_{l=0}^{\infty} h(l)x(k-l) - \sum_{i=0}^{N} w_i(k)x(k-i) \end{aligned} \tag{2.147}$$

where $w_i(k)$ are the coefficients of the adaptive filter.

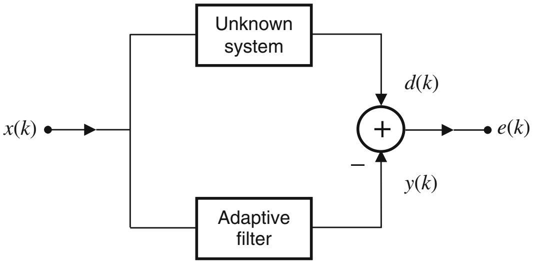

Fig. 2.10 System identification

Assuming that $x(k)$ is a white noise, the MSE for a fixed $\mathbf{w}$ is given by

$$\begin{aligned}\xi &= E\{[\mathbf{h}^T\mathbf{x}_\infty(k) - \mathbf{w}^T\mathbf{x}_{N+1}(k)]^2\}\\ &= E\left[\mathbf{h}^T\mathbf{x}_\infty(k)\mathbf{x}_\infty^T(k)\mathbf{h} - 2\mathbf{h}^T\mathbf{x}_\infty(k)\mathbf{x}_{N+1}^T(k)\mathbf{w} + \mathbf{w}^T\mathbf{x}_{N+1}(k)\mathbf{x}_{N+1}^T(k)\mathbf{w}\right]\\ &= \sigma_x^2\sum_{i=0}^{\infty}h^2(i) - 2\sigma_x^2\mathbf{h}^T\begin{bmatrix}\mathbf{I}_{N+1}\\ \mathbf{0}_{\infty\times(N+1)}\end{bmatrix}\mathbf{w} + \mathbf{w}^T\mathbf{R}_{N+1}\mathbf{w} \qquad (2.148)\end{aligned}$$

where $\mathbf{x}_\infty(k)$ and $\mathbf{x}_{N+1}(k)$ are the input signal vector with infinite and finite lengths, respectively.

By calculating the derivative of ξ with respect to the coefficients of the adaptive filter, it follows that

$$\mathbf{w}_o = \mathbf{h}_{N+1} \qquad (2.149)$$

where

$$\mathbf{h}_{N+1}^T = \mathbf{h}^T\begin{bmatrix}\mathbf{I}_{N+1}\\ \mathbf{0}_{\infty\times(N+1)}\end{bmatrix} \qquad (2.150)$$

If the input signal is a white noise, the best model for the unknown system is a system whose impulse response coincides with the $N + 1$ first samples of the unknown system impulse response. In the cases where the impulse response of the unknown system is of finite length and the adaptive filter is of sufficient order (i.e., it has enough number of parameters), the MSE becomes zero if there is no measurement noise (or channel noise). In practical applications the measurement noise is unavoidable, and if it is uncorrelated with the input signal, the expected value of the adaptive-filter coefficients will coincide with the unknown-system impulse response samples. The output error will of course be the measurement noise. We can observe that the measurement noise introduces a variance in the estimates of the unknown system parameters.

Some real world applications of the system identification scheme include modeling of multipath communication channels [27], control systems [28], seismic exploration [29], and cancellation of echo caused by hybrids in some communication systems [30–34], just to mention a few.

2.10.2 Signal Enhancement

In the signal enhancement application, the reference signal consists of a desired signal $x(k)$ that is corrupted by an additive noise $n_1(k)$. The input signal of the adaptive filter is a noise signal $n_2(k)$ that is correlated with the interference signal $n_1(k)$, but uncorrelated with $x(k)$. Figure 2.11 illustrates the configuration of the signal enhancement application. In practice, this configuration is found in

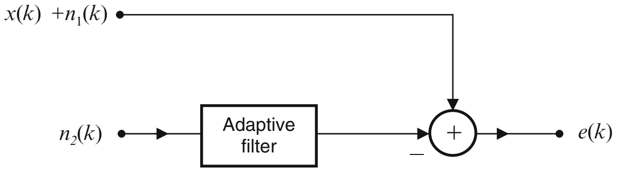

Fig. 2.11 Signal enhancement ($n_1(k)$ and $n_2(k)$ are noise signals correlated with each other)

acoustic echo cancellation for auditoriums [35], hearing aids, noise cancellation in hydrophones [36], cancelling of power line interference in electrocardiography [28], and in other applications. The cancelling of echo caused by the hybrid in some communication systems can also be considered a signal enhancement problem [28].

In this application, the error signal is given by

$$e(k) = x(k) + n_1(k) - \sum_{l=0}^{N} w_l n_2(k-l) = x(k) + n_1(k) - y(k) \tag{2.151}$$

The resulting MSE is then given by

$$E[e^2(k)] = E[x^2(k)] + E\{[n_1(k) - y(k)]^2\} \tag{2.152}$$

where it was assumed that $x(k)$ is uncorrelated with $n_1(k)$ and $n_2(k)$. The above equation shows that if the adaptive filter, having $n_2(k)$ as the input signal, is able to perfectly predict the signal $n_1(k)$, the minimum MSE is given by

$$\xi_{\min} = E[x^2(k)] \tag{2.153}$$

where the error signal, in this situation, is the desired signal $x(k)$.

The effectiveness of the signal enhancement scheme depends on the high correlation between $n_1(k)$ and $n_2(k)$. In some applications, it is useful to include a delay of L samples in the reference signal or in the input signal, such that their relative delay yields a maximum cross-correlation between $y(k)$ and $n_1(k)$, reducing the MSE. This delay provides a kind of synchronization between the signals involved. An example exploring this issue will be presented in the following chapters.

2.10.3 Signal Prediction

In the signal prediction application, the adaptive-filter input consists of a delayed version of the desired signal as illustrated in Fig. 2.12. The MSE is given by

$$\xi = E\{[x(k) - \mathbf{w}^T \mathbf{x}(k-L)]^2\} \tag{2.154}$$

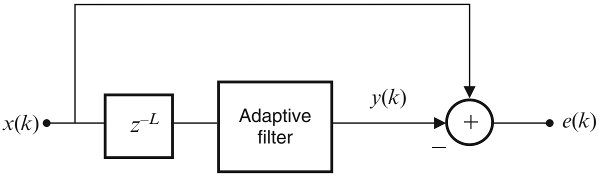

Fig. 2.12 Signal prediction

The minimization of the MSE leads to an FIR filter, whose coefficients are the elements of $\mathbf{w}$. This filter is able to predict the present sample of the input signal using as information old samples such as $x(k-L)$, $x(k-L-1), \ldots, x(k-L-N)$. The resulting FIR filter can then be considered a model for the signal $x(k)$ when the MSE is small. The minimum MSE is given by

$$\xi_{\min} = r(0) - \mathbf{w}_o^T \begin{bmatrix} r(L) \\ r(L+1) \\ \cdot \\ \cdot \\ \cdot \\ r(L+N) \end{bmatrix} \tag{2.155}$$

where $\mathbf{w}_o$ is the optimum predictor coefficient vector and $r(l) = E[x(k)x(k-l)]$ for a stationary process.

A typical predictor's application is in linear prediction coding of speech signals [37], where the predictor's task is to estimate the speech parameters. These parameters $\mathbf{w}$ are part of the coding information that is transmitted or stored along with other information inherent to the speech characteristics, such as pitch period, among others.

The adaptive signal predictor is also used for adaptive line enhancement (ALE), where the input signal is a narrowband signal (predictable) added to a wideband signal. After convergence, the predictor output will be an enhanced version of the narrowband signal.

Yet another application of the signal predictor is the suppression of narrowband interference in a wideband signal. The input signal, in this case, has the same general characteristics of the ALE. However, we are now interested in removing the narrowband interferer. For such an application, the output signal of interest is the error signal [35].

2.10.4 Channel Equalization

As can be seen from Fig. 2.13, channel equalization or inverse filtering consists of estimating a transfer function to compensate for the linear distortion caused

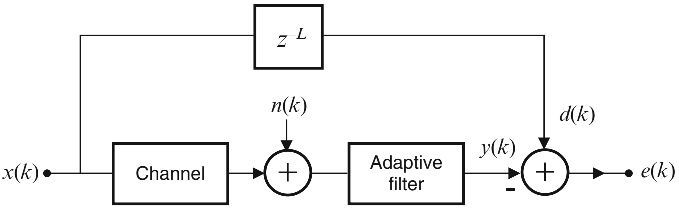

Fig. 2.13 Channel equalization

by the channel. From another point of view, the objective is to force a prescribed dynamic behavior for the cascade of the channel (unknown system) and the adaptive filter, determined by the input signal. The first interpretation is more appropriate in communications, where the information is transmitted through dispersive channels [33,38]. The second interpretation is appropriate for control applications, where the inverse filtering scheme generates control signals to be used in the unknown system [28].

In the ideal situation, where $n(k) = 0$ and the equalizer has sufficient order, the error signal is zero if

$$W(z)H(z) = z^{-L} \tag{2.156}$$

where $W(z)$ and $H(z)$ are the equalizer and unknown system transfer functions, respectively. Therefore, the ideal equalizer has the following transfer function

$$W(z) = \frac{z^{-L}}{H(z)} \tag{2.157}$$

From the above equation, we can conclude that if $H(z)$ is an IIR transfer function with nontrivial numerator and denominator polynomials, $W(z)$ will also be IIR. If $H(z)$ is an all-pole model, $W(z)$ is FIR. If $H(z)$ is an all-zero model, $W(z)$ is an all-pole transfer function.

By applying the inverse $\mathcal{Z}$-transform to (2.156), we can conclude that the optimal equalizer impulse response convolved with the channel impulse response produces as a result an impulse. This means that for zero additional error in the channel, the output signal $y(k)$ restores $x(k - L)$ and, therefore, one can conclude that a deconvolution process took place.

The delay in the reference signal plays an important role in the equalization process. Without the delay, the desired signal is $x(k)$, whereas the signal $y(k)$ will be mainly influenced by old samples of the input signal, since the unknown system is usually causal. As a consequence, the equalizer should also perform the task of predicting $x(k)$ simultaneously with the main task of equalizing the channel. The introduction of a delay alleviates the prediction task, leaving the equalizer free to invert the channel response. A rule of thumb for choosing the delay was proposed and analyzed in [28], where it was conjectured that the best delay should be close to half the time span of the equalizer. In practice, the reader should try different delays.

In the case the unknown system is not of minimum phase, i.e., its transfer function has zeros outside the unit circle of the $\mathcal{Z}$ plane, the optimum equalizer is either stable and noncausal, or unstable and causal. Both solutions are unacceptable. The noncausal stable solution could be better approximated by a causal FIR filter when the delay is included in the desired signal. The delay forces a time shift in the ideal impulse response of the equalizer, allowing the time span, where most of the energy is concentrated, to be in the *causal* region.

If channel noise signal is present and is uncorrelated with the channel's input signal, the error signal and $y(k)$ will be accordingly noisier. However, it should be noticed that the adaptive equalizer, in the process of reducing the MSE, disturbs the optimal solution by trying to reduce the effects of $n(k)$. Therefore, in a noisy environment the equalizer transfer function is not exactly the inverse of $H(z)$.

In practice, the noblest use of the adaptive equalizer is to compensate for the distortion caused by the transmission channel in a communication system. The main distortions caused by the channels are high attenuation and intersymbol interference (ISI). The ISI is generated when different frequency components of the transmitted signals arrive at different times at the receiver, a phenomenon caused by the nonlinear group delay of the channel [38]. For example, in a digital communication system, the time-dispersive channel extends a transmitted symbol beyond the time interval allotted to it, interfering in the past and future symbols. Under severe ISI, when short symbol space is used, the number of symbols causing ISI is large.

The channel impulse response is a time spread sequence described by $h(k)$ with the received signal being given by

$$re(k+J) = x(k)h(J) + \sum_{l=-\infty,\, l\neq k}^{k+J} x(l)h(k+J-l) + n(k+J) \tag{2.158}$$

where J denotes the channel time delay (including the sampler phase). The first term of the above equation corresponds to the desired information, the second term is the interference of the symbols sent before and after $x(k)$. The third term accounts for channel noise. Obviously only the neighboring symbols have significant influence in the second term of the above equation. The elements of the second term involving $x(l)$, for $l > k$, are called pre-cursor ISI since they are caused by components of the data signal that reach the receiver before their cursor. On the other hand, the elements involving $x(l)$, for $l < k$, are called post-cursor ISI.

In many situations, the ISI is reduced by employing an equalizer consisting of an adaptive FIR filter of appropriate length. The adaptive equalizer attempts to cancel the ISI in the presence of noise. In digital communication, a decision device is placed after the equalizer in order to identify the symbol at a given instant. The equalizer coefficients are updated in two distinct circumstances by employing different reference signals. During the equalizer training period, a previously chosen training signal is transmitted through the channel and a properly delayed version of this signal, that is prestored in the receiver end, is used as reference signal. The training signal is usually a pseudo-noise sequence long enough to allow the

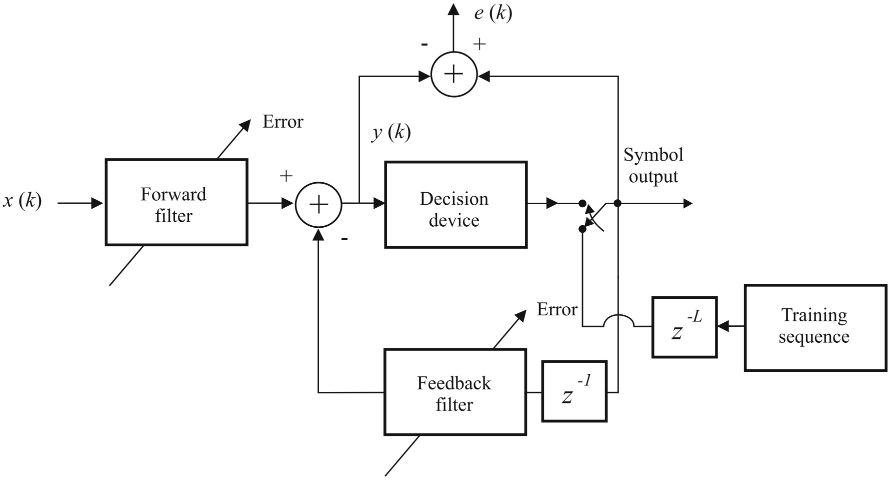

Fig. 2.14 Decision-feedback equalizer

equalizer to compensate for the channel distortions. After convergence, the error between the adaptive-filter output and the decision device output is utilized to update the coefficients. The resulting scheme is the decision-directed adaptive equalizer. It should be mentioned that in some applications no training period is available. Usually, in this case, the decision-directed error is used all the time.

A more general equalizer scheme is the decision-feedback equalizer (DFE) illustrated in Fig. 2.14. The DFE is widely used in situations where the channel distortion is severe [38, 39]. The basic idea is to feed back, via a second FIR filter, the decisions made by the decision device that is applied to the equalized signal. The second FIR filter is preceded by a delay, otherwise there is a delay-free loop around the decision device. Assuming the decisions were correct, we are actually feeding back the symbols $x(l)$, for $l < k$, of (2.158). The DFE is able to cancel the post-cursor ISI for a number of past symbols (depending on the order of the FIR feedback filter), leaving more freedom for the feedforward section to take care of the remaining terms of the ISI. Some known characteristics of the DFE are [38]:

- The signals that are fed back are symbols, being noise free and allowing computational savings.
- The noise enhancement is reduced, if compared with the feedforward-only equalizer.
- Short time recovery when incorrect decisions are made.
- Reduced sensitivity to sampling phase.

The DFE operation starts with a training period where a known sequence is transmitted through the channel, and the same sequence is used at the receiver as the desired signal. The delay introduced in the training signal is meant to compensate for the delay the transmitted signal faces when passing through the channel. During the training period the error signal, which consists of the difference between the delayed training signal and signal $y(k)$, is minimized by adapting the coefficients of the forward and feedback filters. After this period, there is no training signal and the desired signal will consist of the decision device output signal. Assuming the decisions are correct, this *blind* way of performing the adaptation is the best solution to keep track of small changes in the channel behavior.

Example 2.4. In this example we will verify the effectiveness of the Wiener solution in environments related to the applications of noise cancellation, prediction, equalization, and identification.

(a) In a noise cancellation environment a sinusoid is corrupted by noise as follows

$$d(k) = \cos \omega_0 k + n_1(k)$$

with

$$n_1(k) = -an_1(k-1) + n(k)$$

$|a| < 1$ and $n(k)$ is a zero-mean white noise with variance $\sigma_n^2 = 1$. The input signal of the Wiener filter is described by

$$n_2(k) = -bn_2(k-1) + n(k)$$

where $|b| < 1$.

(b) In a prediction case the input signal is modeled as

$$x(k) = -ax(k-1) + n(k)$$

with $n(k)$ being a white noise with unit variance and $|a| < 1$.

(c) In an equalization problem a zero-mean white noise signal $s(k)$ with variance c is transmitted through a channel with an AR model described by

$$\hat{x}(k) = -a\hat{x}(k-1) + s(k)$$

with $|a| < 1$ and the received signal given by

$$x(k) = \hat{x}(k) + n(k)$$

whereas $n(k)$ is a zero-mean white noise with variance d and uncorrelated with $s(k)$.

(d) In a system identification problem a zero-mean white noise signal $x(k)$ with variance c is employed as the input signal to identify an AR system whose model is described by

$$v(k) = -av(k-1) + x(k)$$

where $|a| < 1$ and the desired signal is given by

$$d(k) = v(k) + n(k)$$

Repeat the problem if the system to be identified is an MA whose model is described by

$$v(k) = -ax(k-1) + x(k)$$

For all these cases describe the Wiener solution with two coefficients and comment on the results.

Solution. Some results used in the examples are briefly reviewed. A 2 × 2 matrix inversion is performed as

$$\mathbf{R}^{-1} = \frac{1}{r_{11}r_{22} - r_{12}r_{21}} \begin{bmatrix} r_{22} & -r_{12} \\ -r_{21} & r_{11} \end{bmatrix}$$

where r_{ij} is the element of row i and column j of the matrix $\mathbf{R}$. For two first-order AR modeled signals $x(k)$ and $v(k)$, whose poles are, respectively, placed at $-a$ and $-b$ with the same white-noise input with unit variance, their cross-correlations are given by[9]

$$E[x(k)v(k-l)] = \frac{(-a)^l}{1-ab}$$

for $l > 0$, and

$$E[x(k)v(k-l)] = \frac{(-b)^{-l}}{1-ab}$$

for $l < 0$, are frequently required in the following solutions.

(a) The input signal in this case is given by $n_2(k)$, whereas the desired signal is given by $d(k)$. The elements of the correlation matrix are computed as

$$E[n_2(k)n_2(k-l)] = \frac{(-b)^{|l|}}{1-b^2}$$

The expression for the cross-correlation vector is given by

[9] Assuming $x(k)$ and $v(k)$ are jointly WSS.

$$\begin{aligned}\mathbf{p} &= \begin{bmatrix} E[(\cos\omega_0 k + n_1(k))n_2(k)] \\ E[(\cos\omega_0 k + n_1(k))n_2(k-1)] \end{bmatrix} \\ &= \begin{bmatrix} E[n_1(k)n_2(k)] \\ E[n_1(k)n_2(k-1)] \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{1-ab}\sigma_n^2 \\ -\frac{a}{1-ab}\sigma_n^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{1-ab} \\ -\frac{a}{1-ab} \end{bmatrix}\end{aligned}$$

where in the last expression we substituted $\sigma_n^2 = 1$.

The coefficients corresponding to the Wiener solution are given by

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} = \begin{bmatrix} 1 & b \\ b & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{1-ab} \\ -\frac{a}{1-ab} \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{b-a}{1-ab} \end{bmatrix}$$

The special case where $a = 0$ provides a quite illustrative solution. In this case

$$\mathbf{w}_o = \begin{bmatrix} 1 \\ b \end{bmatrix}$$

such that the error signal is given by

$$\begin{aligned} e(k) &= d(k) - y(k) = \cos\omega_0 k + n(k) - \mathbf{w}_o^T \begin{bmatrix} n_2(k) \\ n_2(k-1) \end{bmatrix} \\ &= \cos\omega_0 k + n(k) - n_2(k) - b n_2(k-1) \\ &= \cos\omega_0 k + n(k) + b n_2(k-1) - n(k) - b n_2(k-1) = \cos\omega_0 k \end{aligned}$$

As can be observed the cosine signal is fully recovered since the Wiener filter was able to restore $n(k)$ and remove it from the desired signal.

(b) In the prediction case the input signal is $x(k)$ and the desired signal is $x(k+L)$. Since

$$E[x(k)x(k-L)] = \frac{(-a)^{|L|}}{1-a^2}$$

the input signal correlation matrix is

$$\begin{aligned}\mathbf{R} &= \begin{bmatrix} E[x^2(k)] & E[x(k)x(k-1)] \\ E[x(k)x(k-1)] & E[x^2(k-1)] \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{1-a^2} & -\frac{a}{1-a^2} \\ -\frac{a}{1-a^2} & \frac{1}{1-a^2} \end{bmatrix}\end{aligned}$$

Vector $\mathbf{p}$ is described by

$$\mathbf{p} = \begin{bmatrix} E[x(k+L)x(k)] \\ E[x(k+L)x(k-1)] \end{bmatrix} = \begin{bmatrix} \frac{(-a)^{|L|}}{1-a^2} \\ \frac{(-a)^{|L+1|}}{1-a^2} \end{bmatrix}$$

The expression for the optimal coefficient vector is easily derived.

$$\begin{aligned} \mathbf{w}_o &= \mathbf{R}^{-1}\mathbf{p} \\ &= (1-a^2)\begin{bmatrix} \frac{1}{1-a^2} & \frac{a}{1-a^2} \\ \frac{a}{1-a^2} & \frac{1}{1-a^2} \end{bmatrix} \begin{bmatrix} \frac{(-a)^L}{1-a^2} \\ \frac{(-a)^{L+1}}{1-a^2} \end{bmatrix} \\ &= \begin{bmatrix} (-a)^L \\ 0 \end{bmatrix} \end{aligned}$$

where in the above equation the value of L is considered positive. The predictor result tells us that an estimate $\hat{x}(k+L)$ of $x(k+L)$ can be obtained as

$$\hat{x}(k+L) = (-a)^L x(k)$$

According to our model for the signal $x(k)$, the actual value of $x(k+L)$ is

$$x(k+L) = (-a)^L x(k) + \sum_{i=0}^{L-1} (-a)^i n(k-i)$$

The results show that if $x(k)$ is an observed data at a given instant of time, the best estimate of $x(k+L)$ in terms of $x(k)$ is to average out the noise as follows

$$\hat{x}(k+L) = (-a)^L x(k) + E\left[\sum_{i=0}^{L-1} (-a)^i n(k-i)\right] = (-a)^L x(k)$$

since $E[n(k-i)] = 0$.

(c) In this equalization problem, matrix $\mathbf{R}$ is given by

$$\mathbf{R} = \begin{bmatrix} E[x^2(k)] & E[x(k)x(k-1)] \\ E[x(k)x(k-1)] & E[x^2(k-1)] \end{bmatrix} = \begin{bmatrix} \frac{1}{1-a^2}c + d & -\frac{a}{1-a^2}c \\ -\frac{a}{1-a^2}c & \frac{1}{1-a^2}c + d \end{bmatrix}$$

By utilizing as desired signal $s(k-L)$ and recalling that it is a white noise and uncorrelated with the other signals involved in the experiment, the cross-correlation vector between the input and desired signals has the following expression

$$\mathbf{p} = \begin{bmatrix} E[x(k)s(k-L)] \\ E[x(k-1)s(k-L)] \end{bmatrix} = \begin{bmatrix} (-1)^L a^L c \\ (-1)^{L-1} a^{L-1} c \end{bmatrix}$$

The coefficients of the underlying Wiener solution are given by

$$\begin{aligned} \mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} &= \frac{1}{\frac{c^2}{1-a^2} + 2\frac{dc}{1-a^2} + d^2} \begin{bmatrix} \frac{1}{1-a^2}c + d & \frac{a}{1-a^2}c \\ \frac{a}{1-a^2}c & \frac{1}{1-a^2}c + d \end{bmatrix} \begin{bmatrix} (-1)^L a^L c \\ (-1)^{L-1} a^{L-1} c \end{bmatrix} \\ &= \frac{(-1)^L a^L c}{\frac{c^2}{1-a^2} + 2\frac{cd}{1-a^2} + d^2} \begin{bmatrix} \frac{c}{1-a^2} + d - \frac{c}{1-a^2} \\ \frac{ac}{1-a^2} - a^{-1}d - \frac{a^{-1}c}{1-a^2} \end{bmatrix} \\ &= \frac{(-1)^L a^L c}{\frac{c^2}{1-a^2} + 2\frac{cd}{1-a^2} + d^2} \begin{bmatrix} d \\ -a^{-1}d - a^{-1}c \end{bmatrix} \end{aligned}$$

If there is no additional noise, i.e., $d = 0$, the above result becomes

$$\mathbf{w}_o = \begin{bmatrix} 0 \\ (-1)^{L-1} a^{L-1} (1 - a^2) \end{bmatrix}$$

that is, the Wiener solution is just correcting the gain of the previously received component of the input signal, namely $x(k-1)$, while not using its most recent component $x(k)$. This happens because the desired signal $s(k-L)$ at instant k has a defined correlation with any previously received symbol. On the other hand, if the signal $s(k)$ is a colored noise the Wiener filter would have a nonzero first coefficient in a noiseless environment. In case there is environmental noise, the solution tries to find a perfect balance between the desired signal modeling and the noise amplification.

(d) In the system identification example the input signal correlation matrix is given by

$$\mathbf{R} = \begin{bmatrix} c & 0 \\ 0 & c \end{bmatrix}.$$

With the desired signal $d(k)$, the cross-correlation vector is described as

$$\mathbf{p} = \begin{bmatrix} E[x(k)d(k)] \\ E[x(k-1)d(k)] \end{bmatrix} = \begin{bmatrix} c \\ -ca \end{bmatrix}$$

The coefficients of the underlying Wiener solution are given by

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} = \begin{bmatrix} \frac{1}{c} & 0 \\ 0 & \frac{1}{c} \end{bmatrix} \begin{bmatrix} c \\ -ca \end{bmatrix} = \begin{bmatrix} 1 \\ -a \end{bmatrix}$$

Note that this solution represents the best way a first-order FIR model can approximate an IIR model, since

$$W_o(z) = 1 - az^{-1}$$

and

$$\frac{1}{1 + az^{-1}} = 1 - az^{-1} + a^2z^{-2} + \cdots$$

On the other hand, if the unknown model is the described FIR model such as $v(k) = -ax(k-1) + x(k)$, the Wiener solution remains the same and corresponds exactly to the unknown system model.

In all these examples, the environmental signals are considered WSS and their statistics assumed known. In a practical situation, not only the statistics might be unknown but the environments are usually nonstationary as well. In these situations, the adaptive filters come into play since their coefficients vary with time according to measured signals from the environment. □

2.10.5 Digital Communication System

For illustration, a general digital communication scheme over a channel consisting of a subscriber line (telephone line, for example) is shown in Fig. 2.15. In either end, the input signal is first coded and conditioned by a transmit filter. The filter shapes the pulse and limits in band the signal that is actually transmitted. The signal then crosses the hybrid to travel through a dual duplex channel. The hybrid is an impedance bridge used to transfer the transmit signal into the channel with minimal leakage to the near-end receiver. The imperfections of the hybrid cause echo that should be properly cancelled.

In the channel, the signal is corrupted by white noise and crosstalk (leakage of signals being transmitted by other subscribers). After crossing the channel and the far-end hybrid, the signal is filtered by the receive filter that attenuates high-frequency noise and also acts as an antialiasing filter. Subsequently, we have a joint DFE and echo canceller, where the forward filter and echo canceller outputs are subtracted. The result after subtracting the decision feedback output is applied to the decision device. After passing through the decision device, the symbol is decoded.

Other schemes for data transmission in subscriber line exist [33]. The one shown here is for illustration purposes, having as special feature the joint equalizer and echo canceller strategy. The digital subscriber line (DSL) structure shown here has been used in integrated services digital network (ISDN) basic access that allows a data rate of 144 Kbits/s [33]. Also, a similar scheme is employed in the high bit rate digital subscriber line (HDSL) [32,40] that operates over short and conditioned

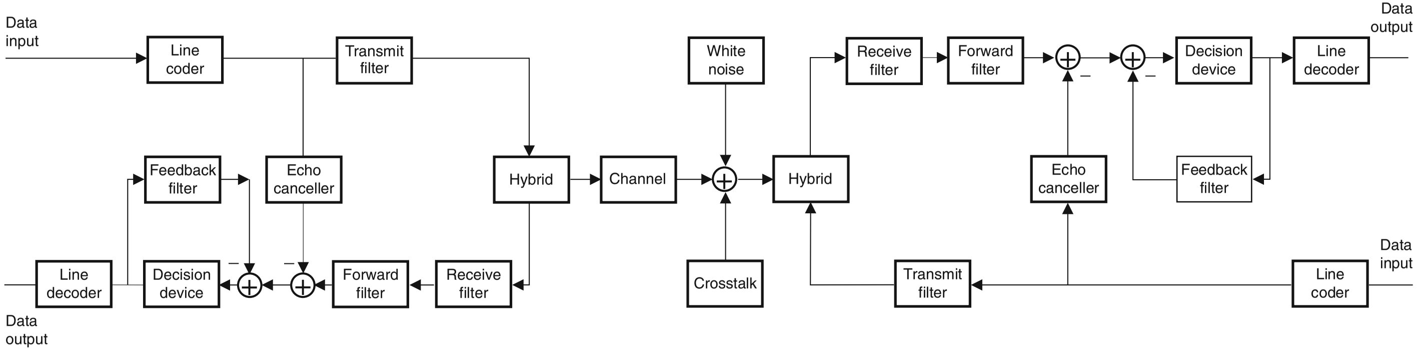

Fig. 2.15 General digital communication transceiver

loops [41, 42]. The latter system belongs to a broad class of digital subscriber line collectively known as XDSL.

In wireless communications, the information is transported by propagating electromagnetic energy through the air. The electromagnetic energy is radiated to the propagation medium via an antenna. In order to operate wireless transmissions, the service provider requires authorization to use a radio bandwidth from government regulators. The demand for wireless data services is more than doubling each year leading to foreseeable spectrum shortage in the years to come. As a consequence, all efforts to maximize the spectrum usage is highly desirable and for sure the adaptive filtering techniques play an important role in achieving this goal. Several examples in the book illustrate how the adaptive filters are employed in many communication systems so that the readers can understand some applications in order to try some new they envision.

2.11 Concluding Remarks

In this chapter, we described some of the concepts underlying the adaptive filtering theory. The material presented here forms the basis to understand the behavior of most adaptive-filtering algorithms in a practical implementation. The basic concept of the MSE surface searching algorithms was briefly reviewed, serving as a starting point for the development of a number of practical adaptive-filtering algorithms to be presented in the following chapters. We illustrated through several examples the expected Wiener solutions in a number of distinct situations. In addition, we presented the basic concepts of linearly constrained Wiener filter required in array signal processing. The theory and practice of adaptive signal processing is also the main subject of some excellent books such as [28, 43–51].

2.12 Problems

1. Suppose the input signal vector is composed by a delay line with a single input signal, compute the correlation matrix for the following input signals:

   (a)
   $$x(k) = \sin\left(\frac{\pi}{6}k\right) + \cos\left(\frac{\pi}{4}k\right) + n(k)$$

   (b)
   $$x(k) = a n_1(k) \cos(\omega_0 k) + n_2(k)$$

   (c)
   $$x(k) = a n_1(k) \sin(\omega_0 k + n_2(k))$$

(d)

$$x(k) = -a_1 x(k-1) - a_2 x(k-2) + n(k)$$

(e)

$$x(k) = \sum_{i=0}^{4} 0.25 n(k-i)$$

(f)

$$x(k) = a n(k) \mathrm{e}^{\jmath \omega_0 k}$$

In all cases, $n(k), n_1(k)$, and $n_2(k)$ are white-noise processes, with zero mean and with variances σ_n^2, $\sigma_{n_1}^2$, and $\sigma_{n_2}^2$, respectively. These random signals are considered independent.

2. Consider two complex random processes represented by $x(k)$ and $y(k)$.
   (a) Derive $\sigma_{xy}^2(k,l) = E[(x(k) - m_x(k))(y(l) - m_y(l))]$ as a function of $r_{xy}(k,l)$, $m_x(k)$ and $m_y(l)$.
   (b) Repeat (a) if $x(k)$ and $y(k)$ are jointly WSS.
   (c) Being $x(k)$ and $y(k)$ orthogonal, in which conditions are they not correlated?

3. For the correlation matrices given below, calculate their eigenvalues, eigenvectors, and conditioning numbers.

   (a)

$$\mathbf{R} = \frac{1}{4} \begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 4 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

   (b)

$$\mathbf{R} = \begin{bmatrix} 1 & 0.95 & 0.9025 & 0.857375 \\ 0.95 & 1 & 0.95 & 0.9025 \\ 0.9025 & 0.95 & 1 & 0.95 \\ 0.857375 & 0.9025 & 0.95 & 1 \end{bmatrix}$$

   (c)

$$\mathbf{R} = 50\sigma_n^2 \begin{bmatrix} 1 & 0.9899 & 0.98 & 0.970 \\ 0.9899 & 1 & 0.9899 & 0.98 \\ 0.98 & 0.9899 & 1 & 0.9899 \\ 0.970 & 0.98 & 0.9899 & 1 \end{bmatrix}$$

(d)

$$\mathbf{R} = \begin{bmatrix} 1 & 0.5 & 0.25 & 0.125 \\ 0.5 & 1 & 0.5 & 0.25 \\ 0.25 & 0.5 & 1 & 0.5 \\ 0.125 & 0.25 & 0.5 & 1 \end{bmatrix}$$

4. For the correlation matrix given below, calculate its eigenvalues and eigenvectors, and form the matrix **Q**.

$$\mathbf{R} = \frac{1}{4}\begin{bmatrix} a_1 & a_2 \\ a_2 & a_1 \end{bmatrix}$$

5. The input signal of a second-order adaptive filter is described by

$$x(k) = \alpha_1 x_1(k) + \alpha_2 x_2(k)$$

where $x_1(k)$ and $x_2(k)$ are first-order AR processes and uncorrelated between themselves having both unit variance. These signals are generated by applying distinct white noises to first-order filters whose poles are placed at a and $-b$, respectively.

(a) Calculate the autocorrelation matrix of the input signal.
(b) If the desired signal consists of $\alpha_3 x_2(k)$, calculate the Wiener solution.

6. The input signal of a first-order adaptive filter is described by

$$x(k) = \sqrt{2}x_1(k) + x_2(k) + 2x_3(k)$$

where $x_1(k)$ and $x_2(k)$ are first-order AR processes and uncorrelated between themselves having both unit variance. These signals are generated by applying distinct white noises to first-order filters whose poles are placed at -0.5 and $\frac{\sqrt{2}}{2}$, respectively. The signal $x_3(k)$ is a white noise with unit variance and uncorrelated with $x_1(k)$ and $x_2(k)$.

(a) Calculate the autocorrelation matrix of the input signal.
(b) If the desired signal consists of $\frac{1}{2}x_3(k)$, calculate the Wiener solution.

7. Repeat the previous problem if the signal $x_3(k)$ is exactly the white noise that generated $x_2(k)$.
8. In a prediction case a sinusoid is corrupted by noise as follows

$$x(k) = \cos\omega_0 k + n_1(k)$$

with

$$n_1(k) = -an_1(k-1) + n(k)$$

where $|a| < 1$. For this case describe the Wiener solution with two coefficients and comment on the results.

9. Generate the ARMA processes $x(k)$ described below. Calculate the variance of the output signal and the autocorrelation for lags 1 and 2. In all cases, $n(k)$ is zero-mean Gaussian white noise with variance 0.1.

   (a)

$$\begin{aligned} x(k) &= 1.9368x(k-1) - 0.9519x(k-2) + n(k) \\ &\quad -1.8894n(k-1) + n(k-2) \end{aligned}$$

   (b)

$$\begin{aligned} x(k) &= -1.9368x(k-1) - 0.9519x(k-2) + n(k) \\ &\quad +1.8894n(k-1) + n(k-2) \end{aligned}$$

   Hint: For white noise generation consult for example [15, 16].

10. Generate the AR processes $x(k)$ described below. Calculate the variance of the output signal and the autocorrelation for lags 1 and 2. In all cases, $n(k)$ is zero-mean Gaussian white noise with variance 0.05.

   (a)

$$x(k) = -0.8987x(k-1) - 0.9018x(k-2) + n(k)$$

   (b)

$$x(k) = 0.057x(k-1) + 0.889x(k-2) + n(k)$$

11. Generate the MA processes $x(k)$ described below. Calculate the variance of the output signal and the autocovariance matrix. In all cases, $n(k)$ is zero-mean Gaussian white noise with variance 1.

   (a)

$$\begin{aligned} x(k) &= 0.0935n(k) + 0.3027n(k-1) + 0.4n(k-2) \\ &\quad + 0.3027n(k-4) + 0.0935n(k-5) \end{aligned}$$

   (b)

$$x(k) = n(k) - n(k-1) + n(k-2) - n(k-4) + n(k-5)$$

   (c)

$$x(k) = n(k) + 2n(k-1) + 3n(k-2) + 2n(k-4) + n(k-5)$$

12. Show that a process generated by adding two AR processes is in general an ARMA process.

13. Determine if the following processes are mean ergodic:

    (a)
    $$x(k) = an_1(k)\cos(\omega_0 k) + n_2(k)$$

    (b)
    $$x(k) = an_1(k)\sin(\omega_0 k + n_2(k))$$

    (c)
    $$x(k) = an(k)\mathrm{e}^{2\jmath\omega_0 k}$$

    In all cases, $n(k), n_1(k)$, and $n_2(k)$ are white-noise processes, with zero mean and with variances σ_n^2, $\sigma_{n_1}^2$, and $\sigma_{n_2}^2$, respectively. These random signals are considered independent.

14. Show that the minimum (maximum) value of (2.69) occurs when $w_i = 0$ for $i \neq j$ and λ_j is the smallest (largest) eigenvalue, respectively.
15. Suppose the matrix **R** and the vector **p** are known for a given experimental environment. Compute the Wiener solution for the following cases:

    (a)
    $$\mathbf{R} = \frac{1}{4}\begin{bmatrix} 4 & 3 & 2 & 1 \\ 3 & 4 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

    $$\mathbf{p} = \begin{bmatrix} \frac{1}{2} & \frac{3}{8} & \frac{2}{8} & \frac{1}{8} \end{bmatrix}^T$$

    (b)
    $$\mathbf{R} = \begin{bmatrix} 1 & 0.8 & 0.64 & 0.512 \\ 0.8 & 1 & 0.8 & 0.64 \\ 0.64 & 0.8 & 1 & 0.8 \\ 0.512 & 0.64 & 0.8 & 1 \end{bmatrix}$$

    $$\mathbf{p} = \frac{1}{4}[0.4096 \;\; 0.512 \;\; 0.64 \;\; 0.8]^T$$

    (c)
    $$\mathbf{R} = \frac{1}{3}\begin{bmatrix} 3 & -2 & 1 \\ -2 & 3 & -2 \\ 1 & -2 & 3 \end{bmatrix}$$

    $$\mathbf{p} = \begin{bmatrix} -2 & 1 & -\frac{1}{2} \end{bmatrix}^T$$

16. For the environments described in the previous problem, derive the updating formula for the steepest-descent method. Considering that the adaptive-filter coefficients are initially zero, calculate their values for the first ten iterations.
17. Repeat the previous problem using the Newton method.
18. Calculate the spectral decomposition for the matrices $\mathbf{R}$ of Problem 15.
19. Calculate the minimum MSE for the examples of Problem 15 considering that the variance of the reference signal is given by σ_d^2.
20. Derive (2.112).
21. Derive the constraint matrix $\mathbf{C}$ and the gain vector $\mathbf{f}$ that impose the condition of linear phase onto the linearly constrained Wiener filter.
22. Show that the optimal solutions of the LCMV filter and the GSC filter with minimum norm are equivalent and related according to $\mathbf{w}_{\text{LCMV}} = \mathbf{T}\mathbf{w}_{\text{GSC}}$, where $\mathbf{T} = [\mathbf{C}\ \mathbf{B}]$ is a full-rank transformation matrix with $\mathbf{C}^T\mathbf{B} = \mathbf{0}$ and

$$\mathbf{w}_{\text{LCMV}} = \mathbf{R}^{-1}\mathbf{C}(\mathbf{C}^T\mathbf{R}^{-1}\mathbf{C})^{-1}\mathbf{f}$$

and

$$\mathbf{w}_{\text{GSC}} = \begin{bmatrix} (\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f} \\ -(\mathbf{B}^T\mathbf{R}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{R}\mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{f} \end{bmatrix}$$

23. Calculate the time constants of the MSE and of the coefficients for the examples of Problem 15 considering that the steepest-descent algorithm was employed.
24. For the examples of Problem 15, describe the equations for the MSE surface.
25. Using the spectral decomposition of a Hermitian matrix show that

$$\mathbf{R}^{\frac{1}{N}} = \mathbf{Q}\boldsymbol{\Lambda}^{\frac{1}{N}}\mathbf{Q}^H = \sum_{i=0}^{N} \lambda_i^{\frac{1}{N}} \mathbf{q}_i \mathbf{q}_i^H$$

26. Derive the complex steepest-descent algorithm.
27. Derive the Newton algorithm for complex signals.
28. In a signal enhancement application, assume that $n_1(k) = n_2(k) * h(k)$, where $h(k)$ represents the impulse response of an unknown system. Also, assume that some small leakage of the signal $x(k)$, given by $h'(k) * x(k)$, is added to the adaptive-filter input. Analyze the consequences of this phenomenon.
29. In the equalizer application, calculate the optimal equalizer transfer function when the channel noise is present.

References

1. D.G. Luenberger, *Introduction to Linear and Nonlinear Programming*, 2nd edn. (Addison Wesley, Reading, 1984)
2. R. Fletcher, *Practical Methods of Optimization*, 2nd edn. (Wiley, New York, 1990)

3. A. Antoniou, W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications* (Springer, New York, 2007)
4. B. Widrow, M.E. Hoff, Adaptive switching circuits. WESCOM Conv. Rec. **4**, 96–140 (1960)
5. B. Widrow, J.M. McCool, M.G. Larimore, C.R. Johnson Jr., Stationary and nonstationary learning characteristics of the LMS adaptive filters. Proc. IEEE **64**, 1151–1162 (1976)
6. A. Papoulis, *Signal Analysis* (McGraw Hill, New York, 1977)
7. A.V. Oppenheim, A.S. Willsky, S.H. Nawab, *Signals and Systems*, 2nd edn. (Prentice Hall, Englewood Cliffs, 1997)
8. P.S.R. Diniz, E.A.B. da Silva, S.L. Netto, *Digital Signal Processing: System Analysis and Design*, 2nd edn. (Cambridge University Press, Cambridge, 2010)
9. A. Antoniou, *Digital Signal Processing: Signals, Systems, and Filters* (McGraw Hill, New York, 2005)
10. L.B. Jackson, *Digital Filters and Signal Processing*, 3rd edn. (Kluwer Academic, Norwell, 1996)
11. R.A. Roberts, C.T. Mullis, *Digital Signal Processing* (Addison-Wesley, Reading, 1987)
12. J.G. Proakis, D.G. Manolakis, *Digital Signal Processing*, 4th edn. (Prentice Hall, Englewood Cliffs, 2007)
13. T. Bose, *Digital Signal and Image Processing* (Wiley, New York, 2004)
14. W.A. Gardner, *Introduction to Random Processes*, 2nd edn. (McGraw Hill, New York, 1990)
15. A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd edn. (McGraw Hill, New York, 1991)
16. P.Z. Peebles Jr., *Probability, Random Variables, and Random Signal Principles*, 3rd edn. (McGraw Hill, New York, 1993)
17. A. Papoulis, Predictable processes and Wold's decomposition: A review. IEEE Trans. Acoust. Speech Signal Process. **ASSP-33**, 933–938 (1985)
18. S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory* (Prentice Hall, Englewood Cliffs, 1993)
19. S.L. Marple Jr., *Digital Spectral Analysis* (Prentice Hall, Englewood Cliffs, 1987)
20. C.R. Johnson Jr., *Lectures on Adaptive Parameter Estimation* (Prentice Hall, Englewood Cliffs, 1988)
21. T. Söderström, P. Stoica, *System Identification* (Prentice Hall International, Hemel Hempstead, Hertfordshire, 1989)
22. G. Strang, *Linear Algebra and Its Applications*, 2nd edn. (Academic, New York, 1980)
23. D.H. Johnson, D.E. Dudgeon, *Array Signal Processing* (Prentice Hall, Englewood Cliffs, 1993)
24. H.L. Van trees, *Optimum Array Processing: Part IV of Detection, Estimation and Modulation Theory* (Wiley, New York, 2002)
25. L.J. Griffiths, C.W. Jim, An alternative approach to linearly constrained adaptive beamforming. IEEE Trans. Antenn. Propag. **AP-30**, 27–34 (1982)
26. M.L.R. de Campos, S. Werner, J.A. Apolinário Jr., Constrained adaptation algorithms employing Householder transformation. IEEE Trans. Signal Process. **50**, 2187–2195 (2002)
27. J.G. Proakis, *Digital Communication*, 4th edn. (McGraw Hill, New York, 2001)
28. B. Widrow, S.D. Stearns, *Adaptive Signal Processing* (Prentice Hall, Englewood Cliffs, 1985)
29. L.C. Wood, S. Treitel, Seismic signal processing. Proc. IEEE **63**, 649–661 (1975)
30. D.G. Messerschmitt, Echo cancellation in speech and data transmission. IEEE J Sel. Areas Comm. **SAC-2**, 283–296 (1984)
31. M.L. Honig, Echo cancellation of voiceband data signals using recursive least squares and stochastic gradient algorithms. IEEE Trans. Comm. **COM-33**, 65–73 (1985)
32. S. Subramanian, D.J. Shpak, P.S.R. Diniz, A. Antoniou, The performance of adaptive filtering algorithms in a simulated HDSL environment, in *Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering*, Toronto, Canada, Sept 1992, pp. TA 2.19.1–TA 2.19.5
33. D.W. Lin, Minimum mean-squared error echo cancellation and equalization for digital subscriber line transmission: Part I – theory and computation. IEEE Trans. Comm. **38**, 31–38 (1990)

34. D.W. Lin, Minimum mean-squared error echo cancellation and equalization for digital subscriber line transmission: Part II – a simulation study. IEEE Trans. Comm. **38**, 39–45 (1990)
35. B. Widrow, J.R. Grover Jr., J.M. McCool, J. Kaunitz, C.S. Williams, R.H. Hearns, J.R. Zeidler, E. Dong Jr., R.C. Goodlin, Adaptive noise cancelling: Principles and applications. Proc. IEEE **63**, 1692–1716 (1975)
36. B.D. Van Veen, K.M. Buckley, Beamforming: A versatile approach to spatial filtering. IEEE Acoust. Speech Signal Process. Mag. **37**, 4–24 (1988)
37. L.R. Rabiner, R.W. Schafer, *Digital Processing of Speech Signals* (Prentice Hall, Englewood Cliffs, 1978)
38. S.U. Qureshi, Adaptive equalization. Proc. IEEE **73**, 1349–1387 (1985)
39. M. Abdulrahman, D.D. Falconer, Cyclostationary crosstalk suppression by decision feedback equalization on digital subscriber line. IEEE J. Sel. Areas Comm. **10**, 640–649 (1992)
40. H. Samueli, B. Daneshrad, R.B. Joshi, B.C. Wong, H.T. Nicholas III, A 64-tap CMOS echo canceller/decision feedback equalizer for 2B1Q HDSL transceiver. IEEE J. Sel. Areas Comm. **9**, 839–847 (1991)
41. J.-J. Werner, The HDSL environment. IEEE J. Sel. Areas Comm. **9**, 785–800 (1991)
42. J.W. Leichleider, High bit rate digital subscriber lines: A review of HDSL progress. IEEE J. Sel. Areas Comm. **9**, 769–784 (1991)
43. M.L. Honig, D.G. Messerschmitt, *Adaptive Filters: Structures, Algorithms, and Applications* (Kluwer Academic, Boston, 1984)
44. S.T. Alexander, *Adaptive Signal Processing* (Springer, New York, 1986)
45. J.R. Treichler, C.R. Johnson Jr., M.G. Larimore, *Theory and Design of Adaptive Filters* (Wiley, New York, 1987)
46. M. Bellanger, *Adaptive Digital Filters and Signal Analysis*, 2nd edn. (Marcel Dekker, Inc., New York, 2001)
47. P. Strobach, *Linear Prediction Theory* (Springer, New York, 1990)
48. B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications* (Wiley, New York, 1998)
49. S. Haykin, *Adaptive Filter Theory*, 4th edn. (Prentice Hall, Englewood Cliffs, 2002)
50. A.H. Sayed, *Fundamentals of Adaptive Filtering* (Wiley, Hoboken, 2003)
51. A.H. Sayed, *Adaptive Filters* (Wiley, Hoboken, 2008)