# 16 CHAPTER

# Adaptive Signal Processing

## LEARNING OBJECTIVES

This chapter deals with different algorithms in adaptive signal processing and its application. After reading this chapter, the reader should be able to:

- Derive the Wiener filter for signals with known second-order statistics.
- Understand steepest descent algorithm, least mean squares (LMS) algorithm and its variants and recursive least squares (RLS) algorithm.
- Apply adaptive algorithm for system identification, signal denoising, prediction, channel equalisation etc.
- Compare adaptive algorithms with respect to performance measures.

## 16.1 INTRODUCTION

Adaptive signal processing is concerned with the design, analysis and implementation of adaptive discrete-time system whose structure adapts itself in response to the external environment or changes in the incoming data. As the name suggests, adaptive signal processing deals with design of digital filter which can adapt against changing external parameters, statistical parameters of the system or disturbances affecting input data. Adaptive signal processing finds application in (i) Echo cancellation (ii) Equalisation of data communication channel (iii) System identification etc. An adaptive system interacts with the environment, learns through the interaction and adjusts its behaviour for optimal performance. An adaptive filter is one whose coefficients vary with respect to time according to certain rules.

## 16.2 MOTIVATION FOR ADAPTIVE FILTER

When the signal of interest and the noise reside in separate frequency bands, conventional linear filters can be used to extract the desired signal. When there is a spectral overlap between signal of interest and noise or whenever the signal or interfering signal's statistics change with time, a fixed coefficient filter will be inappropriate. Again, in real life the signal statistics are not fixed and change with respect to the time. In this situation, there is a need for design of the adaptive filter.

Adaptive filters have the ability to self-adjust their coefficients in order to minimise the error function. The error function is termed as cost function. The cost function has to be minimised according to optimising algorithms. Adaptive filter finds applications in the field of signal processing, communications, radar, sonar, seismology, biomedical engineering, navigation systems etc. The adaptive algorithms are expected to be computationally efficient, numerically robust and fast convergent.

## 16.3 WIENER FILTER

The Wiener filter was introduced by Norbert Wiener in 1949. The Wiener filter is an optimal filter with respect to minimum mean square error (MMSE) sense. The Wiener filter produces an estimate of the desired signal from an observed noisy process assuming that the noise is additive in nature. It is assumed that the signal and the additive noise are stationary in nature. Here it is assumed that the system is a linear time-invariant (LTI) system and the measurements are applied to the input of the LTI system, and the system is designed to produce as its output the MMSE estimate of the process of interest.

### 16.3.1 General framework of the Wiener filter

The general framework of the Wiener filter is illustrated in Figure 16.1.

In Figure 16.1, $s[n]$ represents the original signal which is uncorrupted. The signal $s[n]$ is corrupted by additive noise $\eta[n]$ to yield the signal $x[n]$. The impulse response of the filter is denoted by $h[n]$. The parameters of the filter have to be designed in such a way that the output of the filter $y[n]$ should resemble the desired signal $d[n]$ such that the error '$e[n]$' is minimum. If $h[n]$ is a two-tap linear phase filter with the coefficients $b_0$ and $b_1$, the above structure will be modified as shown in Figure 16.2. The filter structure shown in Figure 16.2 is termed as transversal structure.

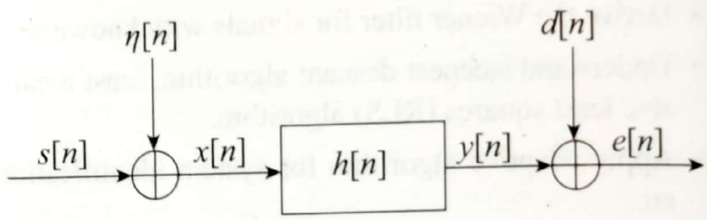


**FIGURE 16.1** General framework of a Wiener filter

From Figure 16.2, the expression for the impulse response of the filter can be given as

$$h[n] = b_0\delta[n] + b_1\delta[n-1] \tag{16.1}$$

The expression for the output signal is obtained as

$$y[n] = x[n] * h[n] \tag{16.2}$$

Substituting expression (16.1) in expression (16.2), the expression for the output is

$$y[n] = b_0 x[n] + b_1 x[n-1] \tag{16.3}$$

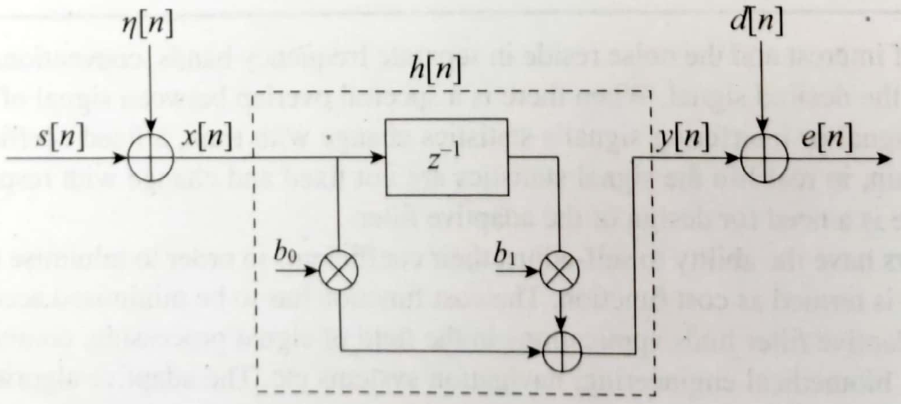


**FIGURE 16.2** Two-tap Wiener filter

Hence this may deduce a framework for the error signal as:

$$e[n] = d[n] - y[n] \tag{16.4}$$

The expression for the squared error can be described by taking the square of the error signal in expression (16.4):

$$e^2[n] = (d[n] - y[n])^2 \tag{16.5}$$

Assuming the input signal to be a statistical stationary signal, we may take the expectation of the squared error to obtain the mean square error as:

$$E\{e^2[n]\} = E\{(d[n] - y[n])^2\} \tag{16.6}$$

In the above expression, '$E$' stands for the expectation operator.
Eq. (16.6) can be expanded as

$$E\{e^2[n]\} = E\{d^2[n] + y^2[n] - 2d[n]y[n]\} \tag{16.7}$$

Substituting expression (16.3) in expression (16.7) we get

$$E\{e^2[n]\} = E\left\{d^2[n] + (b_0x[n] + b_1x[n-1])^2 - 2d[n](b_0x[n] + b_1x[n-1])\right\} \tag{16.8}$$

The above expression can be rewritten as

$$E\{e^2[n]\} = E\left\{\begin{array}{c} d^2[n] + (b_0^2x^2[n] + b_1^2x^2[n-1] + 2b_0b_1x[n]x[n-1]) \\ -2b_0d[n]x[n] - 2b_1d[n]x[n-1] \end{array}\right\} \tag{16.9}$$

Taking the expectation operator inside the expression we get

$$\begin{aligned} E\{e^2[n]\} = E\{d^2[n]\} + b_0^2E\{x^2[n]\} + b_1^2E\{x^2[n-1]\} + 2b_0b_1E\{x[n]x[n-1]\} \\ -2b_0E\{d[n]x[n]\} - 2b_1E\{d[n]x[n-1]\} \end{aligned} \tag{16.10}$$

If $r(l)$ denotes the autocorrelation of $x[n]$ and $p(l)$ denotes the cross-correlation between the desired signal $d[n]$ and the observed signal $x[n]$, the above expression can be rewritten in the reduced form as

$$E\{e^2[n]\} = \sigma_d^2 + b_0^2r(0) + b_1^2r(0) + 2b_0b_1r(1) - 2b_0p(0) - 2b_1p(1) \tag{16.11}$$

The cost function with respect to the filter coefficients $b_0$ and $b_1$ is expressed as

$$J(b_0, b_1) = \sigma_d^2 + b_0^2r(0) + b_1^2r(0) + 2b_0b_1r(1) - 2b_0p(0) - 2b_1p(1) \tag{16.12}$$

The optimal filter coefficient is obtained by minimising the cost function. The cost function can be minimised by taking partial derivative of $J(b_0, b_1)$ with respect to $b_0$ and $b_1$.
Taking partial derivative of $J(b_0, b_1)$ with respect to $b_0$ is given by

$$\frac{\partial J}{\partial b_0} = 2b_0r(0) + 2b_1r(1) - 2p(0) \tag{16.13}$$

Similarly, taking partial derivative of $J(b_0, b_1)$ with respect to $b_1$ is given by

$$\frac{\partial J}{\partial b_1} = 2b_1 r(0) + 2b_0 r(1) - 2p(1) \tag{16.14}$$

The expression for the gradient operator is given from expression (16.13) and (16.14) as

$$\nabla J = \begin{bmatrix} \dfrac{\partial J}{\partial b_0} \\ \dfrac{\partial J}{\partial b_1} \end{bmatrix} = 2\begin{bmatrix} r(0) & r(1) \\ r(1) & r(0) \end{bmatrix}\begin{bmatrix} b_0 \\ b_1 \end{bmatrix} - 2\begin{bmatrix} p(0) \\ p(1) \end{bmatrix} \tag{16.15}$$

In the above expression $b_0$ and $b_1$ are the coefficients of the filter '$h$'. The expression (16.15) in vector form is given by

$$\nabla J = 2Rh - 2p \tag{16.16}$$

To find the optimal filter coefficient the gradient must vanish, hence

$$\nabla J = 0 \tag{16.17}$$

Substituting (16.16) in expression (16.17) we get

$$2Rh_{\text{opt}} = 2p \tag{16.18}$$

From the above expression, the expression for optimal filter is given by

$$h_{\text{opt}} = R^{-1}p \tag{16.19}$$

The above expression is termed as Wiener–Hopf expression which is named after American-born Norbert Wiener and Austrian-born Eberhard Hopf. The expression for optimal filter depends on autocorrelation matrix ($R$) of the observed signal ($x[n]$) and the cross-correlation vector ($p$) between observed signal ($x[n]$) and the desired signal ($d[n]$).

## 16.3.2 Limitations of the Wiener filter

The following are the limitations of the Wiener filter

(i) The autocorrelation matrix of the observed sequence ($R$) and the cross-correlation between the desired and the observed signal ($p$) are not known a priori.
(ii) When the signals are non-stationary, '$R$' and '$p$' change with time.
(iii) The matrix inversion is really challenging.
(iv) Adaptive algorithms are used to compute the solutions which are closer to the optimal solution without computing '$R$' and '$p$' explicitly.

**EXAMPLE 16.1** Signal restoration using the Wiener filter

Write a MATLAB code to restore the signal which is corrupted by white noise. The block diagram which depicts the problem statement is shown in Figure 16.3. In the figure $s[n]$ represents a sinusoidal signal. The frequency of the sinusoidal signal is 5 Hz and the sampling frequency is 100 Hz. This signal $s[n]$ is corrupted by zero mean, unit variance white noise. The objective is to design a Wiener filter which accepts the observed signal $x[n]$ and denoises it. In this case the desired signal is uncorrupted signal $s[n]$. Assume the length of the Wiener filter as nine.

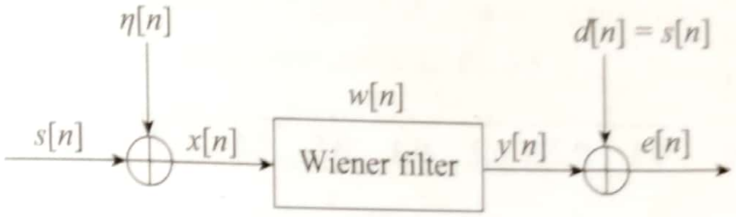


**FIGURE 16.3** Signal restoration using a Wiener filter

## Solution

The expression for the Wiener filter is given by

$$W_{opt} = R^{-1} \times p \tag{16.20}$$

In Eq. (16.20), '$R$' represents the autocorrelation matrix of the observed signal $x[n]$ and '$p$' represents the cross-correlation vector between desired signal and the observed signal. The MATLAB code which implements the Wiener filter is shown in Figure 16.4 and the output is shown in Figure 16.5.

```
%Signal restoration using Wiener filter
clc;clear all;close all;
%Step 1: Generating clean signal s(n)
fs=100;            %Sampling frequency
t=0:1/fs:1;         %Time vector
f1=5;              %Signal frequency
s=sin(2*pi*f1*t);       %clean signal
Ns=length(s);
v=rand(1,Ns); %Step 2: Generation of noise v(n)
x=s+v; %Step 3: Observed signal x(n)
Nh=9; %Step 4: Defining the length of the filter
rxx=xcorr(x); %Step 5: Performing autocorrelation of x(n)
rxx=rxx(Ns:Ns+Nh-1); %Tailoring the ACF of x(n)
rxy=xcorr(s,x);  %Step 6: Performing the cross correlation
rxy=rxy(Ns:Ns+Nh-1);
%Step 7: Wiener filter
w=inv(toeplitz(rxx))*rxy'; %Wopt=inv(R)*p
y=filter(w,1,x); %Filtering the input signal
subplot(2,1,1),plot(t,x),
xlabel('t'),ylabel('Amplitude'),title('Noisy signal')
subplot(2,1,2),plot(t,y)
xlabel('t'),ylabel('Amplitude'),title('Filtered signal')
```

**FIGURE 16.4** MATLAB code to restore the signal using a Wiener filter

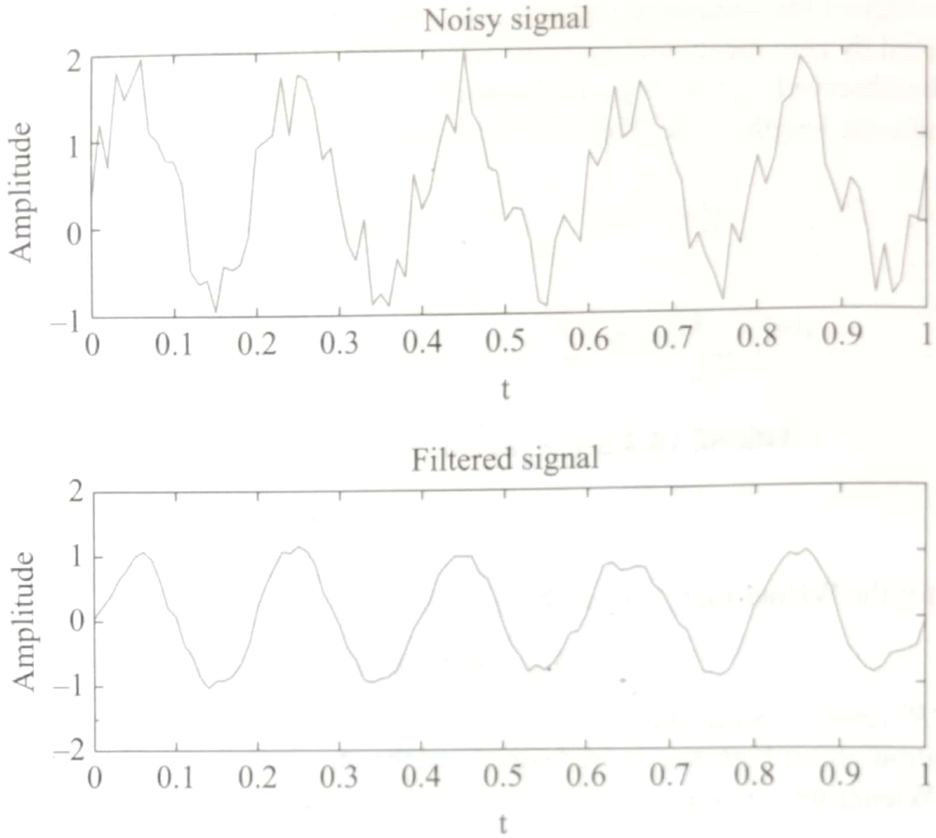


**FIGURE 16.5** Result of MATLAB code shown in Figure 16.4

The built-in functions used in MATLAB code are (i) *sin*() to generate sinusoidal signal (ii) *rand*() to generate white noise which follows uniform distribution (iii) *xcorr*() to perform autocorrelation and cross-correlation operation (iv) *inv*() to compute the inverse of the matrix and (v) *filter*() to obtain the output of the filter. The filter command can also be replaced by *conv*() built-in function.

## EXAMPLE 16.2 Consider the two-tap Wiener filter framework as depicted in Figure 16.6

The expression for the observed sequence is given by $x[n] = s[n] + \eta[n]$. Assume $r_s[0] = 1$ and $r_s[1] = 0.5$ and $r_\eta[n] = \delta[n]$. Both $s[n]$and $\eta[n]$ are wide-sense stationary signals. The cross-correlation between the desired and observed signal is $p = [1, 0.5]^T$. The objective is to design a two-tap optimal filter that will minimise the mean square error between the observed signal and the observed signal.

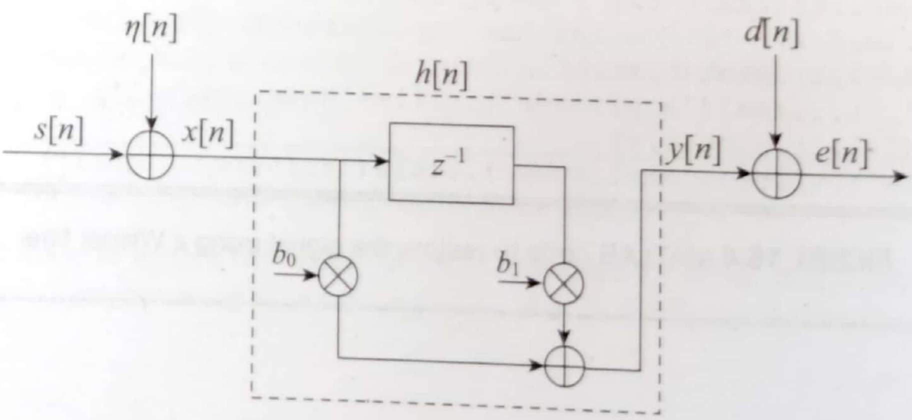


**FIGURE 16.6** Two-tap Wiener filter

**Solution**

Given data

(i) The autocorrelation values of the signal $s[n]$ are $r_s[0]=1$ and $r_s[1]=0.5$.

(ii) The expression for the autocorrelation of the additive noise is $r_\eta[n]=\delta[n]$.

(iii) The expression for the cross-correlation vector is $p=[1, 0.5]^T$.

**To find:** The optimal filter coefficients $b_0$ and $b_1$.

**Formula:** The expression for optimal filter coefficient is given by

$$h_{\text{opt}} = R^{-1}p$$

In the above expression '$R$' represents the autocorrelation of the observed signal.

**Step 1:** To find the autocorrelation matrix '$R$'

From Figure 16.6, the expression for the observed sequence $x[n]$ is given by

$$x[n]=s[n]+\eta[n]$$

From the above expression, we have

$$r_x(l)=r_s(l)+r_\eta(l) \tag{16.21}$$

Substituting $l=0$ in the above expression, we get

$$r_x(0)=r_s(0)+r_\eta(0)$$

Substituting $r_s(0)=1$, $r_\eta(0)=1$ in Eq. (16.21), we get

$$r_x(0)=2$$

Substituting $l=1$ in Eq. (16.21), we get

$$r_x(1)=r_s(1)+r_\eta(1)$$

Substituting $r_s(1)=0.5$, $r_\eta(1)=0$ in the above expression we get

$$r_x(1)=0.5$$

The expression for the $2\times 2$ autocorrelation matrix is given by

$$R=\begin{bmatrix} r_x(0) & r_x(1) \\ r_x(1) & r_x(0) \end{bmatrix}$$

Substituting $r_x(0)=2$ and $r_x(1)=0.5$ in the above expression we get

$$R=\begin{bmatrix} 2 & 0.5 \\ 0.5 & 2 \end{bmatrix}.$$

**Step 2:** To find the expression for optimal filter

The expression for optimal filter is given by

$$h_{\text{opt}} = R^{-1}p$$

Substituting the expression for the autocorrelation matrix and the cross-correlation vector in the above expression we get

$$h_{opt} = \begin{bmatrix} 2 & 0.5 \\ 0.5 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

The expression of optimal filter is given by

$$h_{\text{opt}} = \begin{bmatrix} 0.47 \\ 0.13 \end{bmatrix}$$

The optimal filter coefficients are $b_0 = 0.47$ and $b_1 = 0.13$.

**EXAMPLE 16.3** Consider the block diagram as shown in Figure 16.7. In the figure, $s[n]$ is a sinusoidal signal which is expressed as $s[n] = A\cos(\omega_0 n + \phi)$ with the amplitude $A = 5$ V and $\omega_0 = \frac{\pi}{4}$, the phase angle '$\phi$' is a random variable which follows uniform distribution in the range 0 to $2\pi$. '$v[n]$' represents a white noise zero mean and unit variance. The objective is to design a second-order Wiener filter such that the desired signal $d[n]$ resembles the signal $s[n]$. Assume that the signal and the noise are uncorrelated.

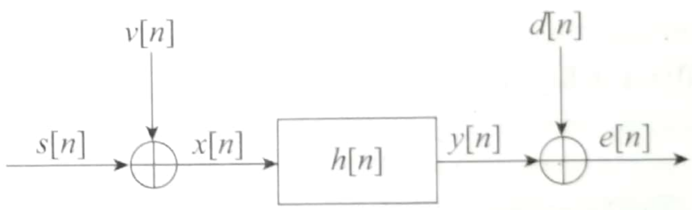


**FIGURE 16.7** Block diagram

**Solution**

Given data

(i) Expression for the signal $s[n]$ is $s[n] = A\cos(\omega_0 n + \phi)$
(ii) The additive noise $v[n]$ has zero mean and unit variance.
(iii) The desired signal is same as the uncorrupted signal $s[n]$. That is $d[n] = s[n]$.

**To find:** (i) Second-order optimal Wiener filter that will minimise the mean square error between the desired signal and the observed signal.

**Step 1:** According to the Wiener–Hopf relation, the expression for optimal filter is given by

$$h_{\text{opt}} = R^{-1} p$$

In the above expression, '$R$' represents the autocorrelation matrix of the observed sequence $x[n]$ and '$p$' denotes the cross-correlation vector between the desired signal and the observed signal. The expression for optimal filter can be given as

$$h_{\text{opt}} = R_{xx}^{-1} \times R_{dx}$$

It is desired to design second-order linear phase filter. We know that second-order filter will have three coefficients, hence the above expression can be written in matrix form as

$$h_{\text{opt}} = \begin{bmatrix} R_{xx}(0) & R_{xx}(1) & R_{xx}(2) \\ R_{xx}(1) & R_{xx}(0) & R_{xx}(1) \\ R_{xx}(2) & R_{xx}(1) & R_{xx}(0) \end{bmatrix}^{-1} \begin{bmatrix} R_{dx}(0) \\ R_{dx}(1) \\ R_{dx}(2) \end{bmatrix}$$

**Step 2:** To find the autocorrelation matrix $R_{xx}$

As per Figure 16.7, the expression for the observed sequence is

$$x[n] = s[n] + v[n]$$

From the above equation, the expression for the autocorrelation of the signal $x[n]$ is given by

$$R_{xx}(k) = R_{ss}(k) + R_{vv}(k) \tag{16.22}$$

**Step 2a:** To find the expression for $R_{ss}(k)$

The expression for the signal $s[n]$ is $s[n] = A\cos(\omega_0 n + \phi)$. The autocorrelation of this signal is given by

$$R_{ss}(k) = E\{s[n]s^*[n-k]\}$$

If the signal is real, the expression for the autocorrelation function is given by

$$R_{ss}(k) = E\{s[n]s[n-k]\}$$

Substituting the expression for $s[n]$ and $s[n-k]$ in the above expression we get

$$R_{ss}(k) = E\{A\cos(\omega_0 n + \phi)A\cos(\omega_0(n-k) + \phi)\}$$

Simplifying the above expression, the expression for the autocorrelation is given by

$$R_{ss}(k) = \frac{A^2}{2}\cos(\omega_0 k) \tag{16.23}$$

**Step 2b:** To find the expression for $R_{vv}(k)$

The signal $v[n]$ is an additive noise with zero mean and unit variance. The expression for the autocorrelation of white noise is given by

$$R_{vv}(k) = \sigma_v^2 \delta(k) \tag{16.24}$$

It is mentioned in the problem statement that the variance value is unity; hence the above expression is given by

$$R_{vv}(k) = \delta(k)$$

Combining the results of step 2a and step 2b, the expression for autocorrelation of observed sequence is given by

$$R_{xx}(k) = \frac{A^2}{2}\cos(\omega_0 k) + \delta(k) \tag{16.25}$$

Substituting $k = 0$ in the above expression we get

$$R_{xx}(0) = \frac{A^2}{2}\cos(\omega_0 \times 0) + \delta(0)$$

Substituting cos(0) = 1 and δ(0) = 1 in the above expression we get

$$R_{xx}(0) = \frac{A^2}{2} + 1 \tag{16.26}$$

Substituting $k = 1$ in the above expression (16.25) we get

$$R_{xx}(1) = \frac{A^2}{2}\cos(\omega_0) + \delta(1)$$

Substituting $\delta(1) = 0$ in the above expression we get

$$R_{xx}(1) = \frac{A^2}{2}\cos(\omega_0) \tag{16.27}$$

Substituting $k = 2$ in the above expression (16.25) we get

$$R_{xx}(2) = \frac{A^2}{2}\cos(2\omega_0) + \delta(2)$$

Substituting $\delta(2) = 0$ in the above expression we get

$$R_{xx}(2) = \frac{A^2}{2}\cos(2\omega_0) \tag{16.28}$$

Substituting $A = 5\text{V}$ and $\omega_0 = \frac{\pi}{4}$ in Eqs (16.26)–(16.28), the values of $R_{xx}(0)$, $R_{xx}(1)$ and $R_{xx}(2)$ are obtained as $R_{xx}(0) = 13.5$, $R_{xx}(1) = \frac{12.5}{\sqrt{2}}$ and $R_{xx}(2) = 0$.

**Step 3:** To obtain the expression for cross-correlation vector $R_{dx}(k)$

The expression for the cross-correlation vector $R_{dx}(k)$ is given by

$$R_{dx}(k) = E\{d[k]x^*[n-k]\}$$

For a real signal, the above expression can be written as

$$R_{dx}(k) = E\{d[k]x[n-k]\} \tag{16.29}$$

From Eq. (16.21), $x[n] = s[n] + v[n]$, this implies

$$x[n-k] = s[n-k] + v[n-k] \tag{16.30}$$

Substituting Eq. (16.30) in Eq. (16.29) we get

$$R_{dx}[k] = E\left\{d[k]\left[s[n-k] + v[n-k]\right]\right\} \tag{16.31}$$

The above expression can be written as

$$R_{dx}(k) = E\left\{d[k]s[n-k]\right\} + E\left\{d[k]v[n-k]\right\} \tag{16.32}$$

It is mentioned in the problem statement that $d[n] = s[n]$, hence the above expression can be written as

$$R_{dx}(k) = E\{s[k]s[n-k]\} + E\{s[k]v[n-k]\} \tag{16.33}$$

Since the signal is independent of the noise, we have

$$E\{s[k]v[n-k]\} = 0 \tag{16.34}$$

Substituting Eq. (16.34) in Eq. (16.33) we get

$$R_{dx}(k) = E\{s[k]s[n-k]\}$$

From the definition of autocorrelation, the above expression can be written as

$$R_{dx}(k) = R_{ss}(k)$$

where $R_{ss}(k) = \dfrac{A^2}{2}\cos(\omega_0 k)$, hence

$$R_{dx}(k) = \frac{A^2}{2}\cos(\omega_0 k) \tag{16.35}$$

Substituting $A = 5$ and $\omega_0 = \dfrac{\pi}{4}$ in Eq. (16.35) we get

$$R_{dx}(k) = 12.5\cos\left(\frac{\pi}{4}k\right) \tag{16.36}$$

Substituting $k = 0$ in the above equation we get $R_{dx}(0) = 12.5$. Substituting $k = 1$, the value of $R_{dx}(1) = \dfrac{12.5}{\sqrt{2}}$ and substituting $k = 2$ we get $R_{dx}(2) = 0$.

**Step 4:** To find the optimal filter coefficient

Substituting the expression of autocorrelation matrix from step 2 and the cross-correlation vector from step 3, the value of optimal filter coefficient is obtained as

$$h_{\text{opt}} = \begin{bmatrix} 13.5 & \frac{13.5}{\sqrt{2}} & 0 \\ \frac{13.5}{\sqrt{2}} & 13.5 & \frac{13.5}{\sqrt{2}} \\ 0 & \frac{13.5}{\sqrt{2}} & 13.5 \end{bmatrix}^{-1} \begin{bmatrix} 12.5 \\ \frac{12.5}{\sqrt{2}} \\ 0 \end{bmatrix}$$

Solving the above expression, the optimal filter coefficient is computed as

$$h_{\text{opt}} = \begin{bmatrix} 0.707 \\ 0.34 \\ -0.226 \end{bmatrix}$$

Scanned with OKEN Scanner