9. Digital Signal Processing

Nikolaos I. Xiros

In this chapter the concept of discrete-time signals and sampled-data systems implemented on digital hardware versus those of continuous-time signals driving analog systems and processes is introduced early on. The processes of signal sampling and analog signal reconstruction are then investigated, and the Nyquist sampling rate to avoid aliasing is explained by means of Fourier series analysis. Then the Z-transform is introduced as the tool of preference for analysis and synthesis of discrete-time linear, time-invariant systems defined by difference equations in the discrete time domain. A detailed account of the most important and practical continuous-time system mapping techniques to discrete-time ones is then presented. A brief account of digital filter structures and types is also given along with a presentation of the fast Fourier transform algorithm for the calculation of the discrete Fourier transform of discrete-time signals with finite duration admitting periodic expansion. The notions of waveform statistics as encountered in random signals and stochastic processes are then given in order to conclude with the effect on a random signal's statistics due to its propagation through a linear, time-invariant system. Finally, concepts of optimal signal estimation and the Wiener filter are presented, leading to matched filter and zero-forcing equalizer parametric designs in the discrete-time domain.

9.1 Discrete-Time Systems

The progress in integrated circuit manufacturing (Very Large Scale Integration (VLSI)) combined with the need for high complexity and accuracy signal analysis and processing algorithms without development cost or time increase are the basic factors that contributed to the continuously increasing use of digital systems in industry and technology [9.1, 2].

9.1.1 Discrete-Time Signals and Digital Systems

A digital processing system is a system where the basic mathematical operations and functions, including signal transformations and data series analysis, are implemented as an embedded program onboard real-time, computer hardware [9.1, 2]. Since the computer hardware employed is digital, the signals that can be processed or generated cannot be continuous time, i. e., defined for any instant in time. Instead, the computer processes and generates tables (i. e., *vectors*) of values for the input and output signal, respectively. Such tables hold the values of the corresponding signals at discrete time instants and are referred to as discrete-time signals. The time instants for which a discrete-time signal is known or defined are in most practical cases equally spaced; this scheme is known as uniform sampling and the spacing interval between two successive samples of the signal is mentioned as sampling interval or period.

The systems, e.g., real-time digital hardware, processing discrete-time signals are known as discrete-time systems. The main advantages in using computers in marine vehicle or process instrumentation, telecommunications, acoustics etc., instead of analog (continuous-time) systems are [9.1, 3–5]:

1. (Re)use of general purpose hardware for implementing a wide variety of algorithms instead of custom-built, proprietary designs that may become obsolete causing higher acquisition and development cost.
2. Changes in the algorithm can be easily implemented as changes in the software through reprogramming without need to modify the hardware.
3. Easy implementation of adaptive or time-varying algorithms due to programming.
4. Very small to zero sensitivity to environmental conditions, e.g., ambient temperature, in contrast to, e.g., analog electric circuitry.
5. Given, well-defined accuracy in computation determined by the computer's word length.
6. User-friendly human–machine interface (HMI).
7. Network connectivity.

However, there are some disadvantages in using discrete-time systems, the most important of which is the need for converter or adapter circuits or interface systems at their input and output in order to be able to interact with the vast majority of the real-world physical or engineering systems or processes, which are most often continuous-time. Such interfaces are the analog-to-digital (A/D) and digital-to-analog (D/A) converters.

9.1.2 Signal Sampling

Modeling of Ideal Sampler

Consider a continuous-time signal $x(t)$, which is driven through a sampling circuit (sampler) operating at a constant sampling rate (frequency), $f_\mathrm{s} = 1/T_\mathrm{s}$, as shown in Fig. 9.1.

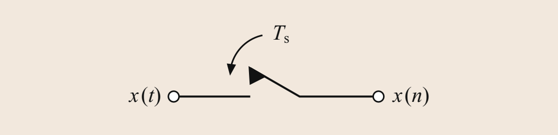


Fig. 9.1 Analog signal sampler

The sampler's output is a sequence of entries of the sampled signal, $x(nT_\mathrm{s})$, where n is an integer and T_s the sampling period. Sequence $x(n) = x(nT_\mathrm{s})$ is a discrete-time signal.

We will now represent the discrete-time signal as well as the sampling process in a continuous-time framework. Such a representation can be derived on the basis of Fig. 9.2.

A pulsed continuous-time signal $x_\Delta(t)$, which corresponds to sequence $x(nT_\mathrm{s})$, can be generated from the original continuous-time signal $x(t)$ (dashed line), if the latter is multiplied by the following *sampling pulse train* $P_\Delta(t)$

$$P_\Delta(t) = \sum_{n=-\infty}^{+\infty} p_\Delta(t - nT_\mathrm{s}) \,, \tag{9.1}$$

where

$$p_\Delta(t) = \begin{cases} 1/\Delta, & t \in [0, \Delta] \\ 0 & t \notin [0, \Delta] \end{cases} \,, \quad 0 < \Delta \ll T_\mathrm{s} \,. \tag{9.2}$$

In effect

$$x_\Delta(t) = x(t) \cdot P_\Delta(t) \,. \tag{9.3}$$

However, the following holds

$$\lim_{\Delta\to 0} p_\Delta(t) = \delta(t) \Rightarrow \lim_{\Delta\to 0} P_\Delta(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT_\mathrm{s}) \,. \tag{9.4}$$

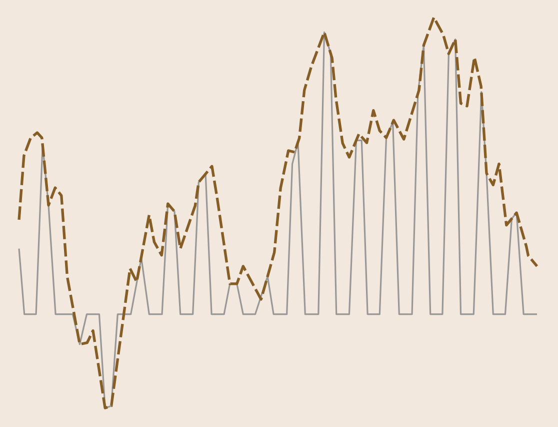


Fig. 9.2 Sampling of continuous-time waveform at uniform rate

In the above *ideal* case, the sampling pulse train is an impulse train or Dirac comb. In effect, (9.4) becomes

$$\begin{aligned} x_\delta(t) &= x(t)\cdot \sum_{n=-\infty}^{+\infty} \delta(t-nT_\mathrm{s}) \\ &= \sum_{n=-\infty}^{+\infty} x(nT_\mathrm{s})\delta(t-nT_\mathrm{s}) \\ &= \sum_{n=-\infty}^{+\infty} x(n)\delta(t-nT_\mathrm{s}) \,. \end{aligned} \tag{9.5}$$

Above, the following fact for Dirac's impulse delta has been used: $\chi(t)\delta(t) = \chi(0)\delta(t)$, which implies the assumption that signal $\chi(t)$ assumes non-infinite values.

Hence, the ideal sampler operation consists of multiplicatively applying the input signal by a Dirac comb with T_s. Now we will determine the spectrum of signal $x_\delta(t)$ at the output of the sampler by employing the Fourier transform.

Fourier Series Expansion and Fourier Transform

Any periodic signal $x(t)$ with period T can be expanded to a complex Fourier series according to the following

$$x(t) = \sum_{n=0}^{+\infty} c_n \exp\left(in\frac{2\pi}{T}t\right) . \tag{9.6}$$

Coefficients c_n of the series are calculated according to

$$c_n = \frac{1}{T}\int_0^T x(t)\exp\left(-in\frac{2\pi}{T}t\right)\mathrm{d}t \,. \tag{9.7}$$

The above can be extended to the case of aperiodic (non-periodic) signals assuming that period T is infinite. That is how the *Fourier transform* emerges

$$\begin{aligned} X(\omega) &= \mathcal{F}\{x(t)\} \\ &= \int_{-\infty}^{+\infty} x(t)\exp(-i\omega t)\,\mathrm{d}t, \quad \omega = \frac{2\pi}{T} \,. \end{aligned} \tag{9.8}$$

The inverse Fourier transform is defined by

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} X(\omega)\exp(i\omega t)\,\mathrm{d}\omega \,. \tag{9.9}$$

To generalize formalism we refer to the Fourier transform of any signal, periodic or not. In the case of periodic signals, the Fourier transform is directly connected to coefficients c_n of its series expansion as follows

$$X(\omega) = 2\pi\sum_{n=0}^{+\infty} c_n\delta\left(\omega - n\frac{2\pi}{T}\right) . \tag{9.10}$$

The above can be derived directly by plugging it into (9.8) and generating (9.6). The Fourier transform, just like the Laplace transform, from which the former can be defined by setting $s = i\omega$, is a linear integral transform. Its properties are identical to those of the Laplace transform, and can be sought in the literature. An important property is that of multiplication-convolution duality

$$\begin{aligned} x(t) &= x_1(\xi)\otimes x_2(\xi) \\ &\triangleq \int_{-\infty}^{+\infty} x_1(\xi)\cdot x_2(t-\xi)\cdot \mathrm{d}\xi \\ &\Rightarrow X(\omega) = X_1(\omega)\cdot X_2(\omega) \\ x(t) &= x_1(t)\cdot x_2(t) \Rightarrow X(\omega) \\ &= X_1(\omega)\otimes X_2(\omega) \\ &\triangleq \int_{-\infty}^{+\infty} X_1(\xi)\cdot X_2(\omega-\xi)\cdot \mathrm{d}\xi \,. \end{aligned} \tag{9.11}$$

Here, $\otimes$ stands for the convolution operator. Note that the above can be directly applied to the Dirac comb describing the function of an ideal sampler as well as its periodicity.

Indeed, the Dirac comb in (9.4) is a periodic signal with period T_s. Its Fourier series is given below

$$\begin{aligned} c_n &= \frac{1}{T_\mathrm{s}}\int_0^{T_\mathrm{s}} \delta(t)\exp(-in\omega_\mathrm{s}t)\,\mathrm{d}t \\ &= \frac{\exp(-in\omega_\mathrm{s}0)}{T_\mathrm{s}}\int_0^{T_\mathrm{s}}\delta(t)\,\mathrm{d}t = \frac{1}{T_\mathrm{s}}, \quad \omega_\mathrm{s} = \frac{2\pi}{T_\mathrm{s}} \,. \end{aligned} \tag{9.12}$$

Hence, based on (9.10) the spectrum (Fourier transform) of the Dirac comb is given by

$$\begin{aligned} \mathcal{F}\left\{\sum_{n=-\infty}^{+\infty}\delta(t-nT_\mathrm{s})\right\} &= 2\pi\sum_{n=0}^{+\infty}\frac{1}{T_\mathrm{s}}\delta\left(\omega - n\frac{2\pi}{T_\mathrm{s}}\right) \\ &= \omega_\mathrm{s}\sum_{n=0}^{+\infty}\delta(\omega - n\omega_\mathrm{s}) \,. \end{aligned} \tag{9.13}$$

Nyquist Sampling Rate – Aliasing

Combining the relationship above for the Dirac comb spectrum with (9.5), as well as the Fourier transform du-

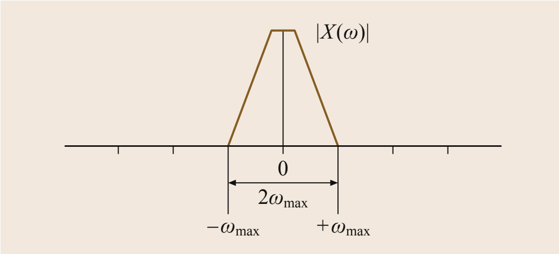


Fig. 9.3 Typical spectrum of real low-pass signal

ality between multiplication and convolution, one can determine the spectrum of signal $x_\delta(t)$ at the output of an ideal sampler

$$
\begin{aligned}
x_\delta(t) &= x(t) \cdot \sum_{n=-\infty}^{+\infty} \delta(t - nT_\mathrm{s}) \Rightarrow \\
&\Rightarrow X_\delta(\omega) = X(\omega) \otimes \mathcal{F}\left\{ \sum_{n=-\infty}^{+\infty} \delta(t - nT_\mathrm{s}) \right\} \\
&= \omega_\mathrm{s} X(\omega) \otimes \sum_{n=0}^{+\infty} \delta(\omega - n\omega_\mathrm{s}) \\
&\Rightarrow X_\delta(\omega) = \omega_\mathrm{s} \sum_{n=0}^{+\infty} X(\omega) \otimes \delta(\omega - n\omega_\mathrm{s}) \\
&= \omega_\mathrm{s} \sum_{n=0}^{+\infty} \int_{-\infty}^{+\infty} X(\xi)\delta(\omega - n\omega_\mathrm{s} - \xi)\,\mathrm{d}\xi \,.
\end{aligned}
$$

The following property of the Dirac impulse delta can now be employed

$$
\begin{aligned}
&\chi(r) \otimes \delta(r - r_0) \\
&\quad = \int_{-\infty}^{+\infty} \chi(\xi)\delta(r - r_0 - \xi)\,\mathrm{d}\xi = \chi(r - r_0) \,.
\end{aligned}
\tag{9.14}
$$

Eventually one obtains

$$
\begin{aligned}
X_\delta(\omega) &= \mathcal{F}\left\{ \sum_{n=-\infty}^{+\infty} x(nT_\mathrm{s})\delta(t - nT_\mathrm{s}) \right\} \\
&= \omega_\mathrm{s} \cdot \sum_{n=0}^{+\infty} X(\omega - n\omega_\mathrm{s}) \,.
\end{aligned}
\tag{9.15}
$$

Using (9.15) one can assess the effect of sampling on spectrum $X(\omega)$ of the analog, the continuous-time signal at the input. Sampling introduces, therefore, an infinite number of aliases of spectrum $X(\omega)$ of the analog signal, centered at integer multiples of the sampling circular frequency ω_s.

To visualize this, consider a real signal with all its spectral content lying within the low frequency range (low-pass or baseband signal); like, e.g., the signal with spectrum as in Fig. 9.3. Note that the amplitude spectrum $|X(\omega)|$ of the signal is an odd function of circular frequency ω due to the fact is assumed to be purely real in the time domain.

The amplitude spectrum of signal $x_\delta(t)$ is shown in the Fig. 9.4 as generated by sampling of low-pass signal $x(t)$ above at three (9.3) different sampling rates: $\omega_{\mathrm{s},0} = 2\omega_{\max}$, $\omega_{\mathrm{s},1} = 4\omega_{\max}$, $\omega_{\mathrm{s},2} = \omega_{\max}$.

Using the plot, one can readily derive the following sampling rate criterion attributed to Nyquist (or the Shannon sampling theorem) for the low-pass signal like the one in Fig. 9.4

$$
\omega_\mathrm{s} \geq \omega_{\mathrm{s},0} = 2\omega_{\max} \,. \tag{9.16}
$$

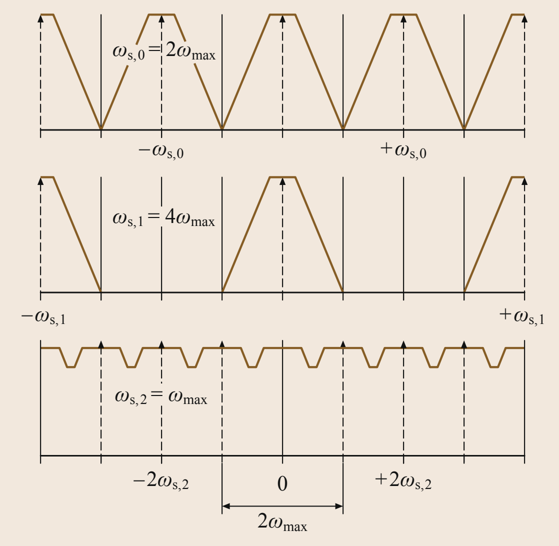
If the condition above holds as seen in (9.15) (spectrum periodicity for the sampled signal), as well as Fig. 9.5, an ideal (*brick*) low-pass filter with transfer function as follows can be used to reconstruct the original continuous-time signal

$$
H_{\mathrm{LPF}}(\omega) = \begin{cases} 1, & |\omega| \leq \omega_\mathrm{s}/2 \\ 0, & |\omega| > \omega_\mathrm{s}/2 \end{cases} \,. \tag{9.17}
$$

The original spectrum is ideally reproduced without any distortion since it holds that

$$
H_{\mathrm{LPF}}(\omega) X_\delta(\omega) = \omega_\mathrm{s} X(\omega) \,. \tag{9.18}
$$

This is made clearer next when reconstruction of a continuous-time signal based on the values of a discretized version is discussed.



$\omega_{\mathrm{s},0} = 2\omega_{\max}$
$-\omega_{\mathrm{s},0}$
$+\omega_{\mathrm{s},0}$
$\omega_{\mathrm{s},1} = 4\omega_{\max}$
$-\omega_{\mathrm{s},1}$
$+\omega_{\mathrm{s},1}$
$\omega_{\mathrm{s},2} = \omega_{\max}$
$-2\omega_{\mathrm{s},2}$
0
$+2\omega_{\mathrm{s},2}$
$2\omega_{\max}$

Fig. 9.4 Sampled signal spectrum at various sampling rates

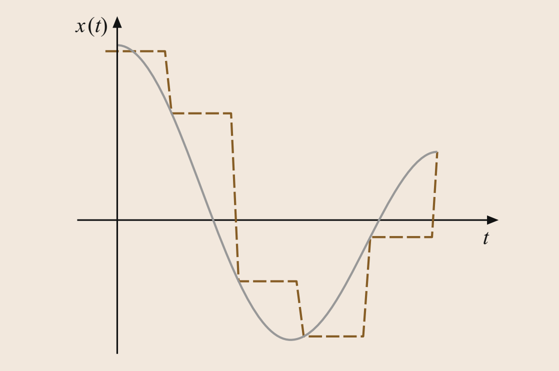


Fig. 9.5 Original continuous-time signal (*continuous line*) and reconstructed version using a non-brick filter (*dashed line*)

9.1.3 Analog Signal Reconstruction Using a Discrete-Time Signal

The simplest process to generate a continuous-time signal, $x(t)$, from a discretized sequence (discrete-time signal) $x(n)$ is to employ the zero-order hold (ZOH) network. This system *holds* the output constant and equal to the one applied to its input for an entire sampling period T_s. In effect, the output of ZOH when driven by a discretized signal is given in the Fig. 9.5.

The mathematical description of a ZOH system is achieved through the following impulse response and transfer function in the complex frequency domain of variable s

$$\begin{aligned} h_{\text{ZOH}}(t) &= u_{\text{step}}(t) - u_{\text{step}}(t - T_s) \\ \Leftrightarrow H_{\text{ZOH}}(s) &= \frac{1 - e^{-sT_s}}{s} . \end{aligned} \tag{9.19}$$

To better understand the action of the ZOH system, its transfer function in the circular frequency domain ω (Fourier transform) is given below, as well

$$\begin{aligned} H_{\text{ZOH}}(\omega) &= \frac{1 - e^{-i\omega T_s}}{i\omega} = T_s e^{-i\omega T_s/2} \frac{\sin(\omega T_s/2)}{(\omega T_s/2)} \\ &= \frac{e^{-i\pi f/f_s}}{f_s} \operatorname{sinc}(f/f_s) . \end{aligned} \tag{9.20}$$

In the above $\operatorname{sinc}(\chi) = \operatorname{sinc}(\pi\chi)/\pi\chi$. In Fig. 9.6 the amplitude spectra of the transfer functions of the ZOH system (normalized by setting $f_s = 1$ Hz) and the ideal reconstruction low-pass filter which was introduced with (9.17).

Indeed, in the circuit above if resistance R is small (theoretically zero), then time constant τ indicating the time needed for charging the capacitor in the hold circuit is small when compared to sampling period T_s. The voltage across the terminals of the capacitor stays at the value set in the previous sampling instant, when the switch was momentarily switched on, until the next instant the switch is on, etc. The differences of the analog memory element, as the switched resistor and capacitor (RC) circuit is known, and the model ZOH system are:

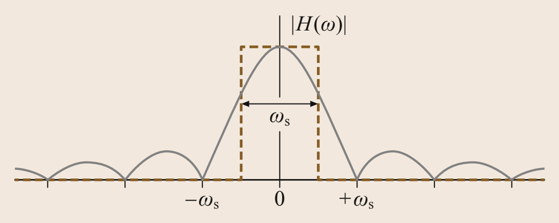


Fig. 9.6 Amplitude spectrum of transfer function $H_{\text{ZOH}}(\omega)$ (*continuous line*) and $H_{\text{LPF}}(\omega)$ (*dashed line*)

a) The ohm resistance value of the circuit is non-zero and, therefore, the time to charge the capacitor is finite and not zero.
b) The duration of the conduction phase of the switch is finite and non-zero.
c) The ohm resistance applied to the terminals of the capacitor when the switch is off is not infinite (open circuit) but finite, eventually leading to discharge of the capacitor.

The use of the RC hold circuit is widespread due to its simplicity. However, if a more accurate reconstruction with less distortion, without increasing the sampling rate, then hold circuits of order higher than zero (e.g., first-order hold, FOH) so that its transfer function better approximates the ideal *brick* low-pass filter in (9.17), as shown in Fig. 9.6.

9.1.4 The *Z*-Transform

The single-sided (unilateral) Z-transform is a linear transform defined as follows for a discrete-time signal $x(n)$

$$X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=0}^{+\infty} x(n) \cdot z^{-n} . \tag{9.21}$$

The Z-transform is derived from the Laplace transform of signal $x_\delta(t)$, which is generated by the ideal sampler shown in Fig. 9.7,

$$\begin{aligned} x_\delta(t) &= \sum_{n=-\infty}^{+\infty} x(n)\delta(t - nT_s) \Rightarrow X_\delta(s) \\ &= \mathcal{L}\left\{\sum_{n=-\infty}^{+\infty} x(n)\delta(t - nT_s)\right\} = \sum_{n=0}^{+\infty} x(n) e^{-n \cdot sT_s} . \end{aligned} \tag{9.22}$$

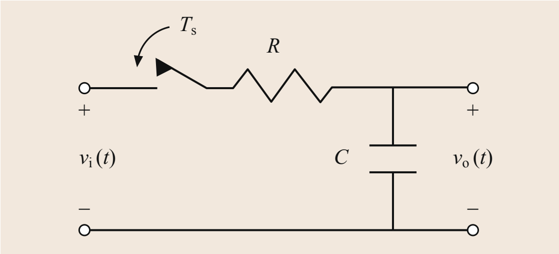


Fig. 9.7 Switched RC circuit modeled by the ZOH system

In effect,

$$X(z)|_{z^{-1}=\exp(-sT_s)} = X_\delta(s) . \tag{9.23}$$

Variable z^{-1} is used for describing a signal, or system as we will see later on. Its significance lies in the fact that multiplying by z^{-1} in the transform domain translates to a time shift by one sampling period, T_s, in the discrete-time domain n. Indeed,

$$y(n) = x(n-1) \Leftrightarrow Y(z) = z^{-1}X(z) . \tag{9.24}$$

Furthermore, here are some more of the most important properties of the Z-transform:

1. Linearity

$$\left.\begin{aligned} X_1(z) &= \mathcal{Z}\{x_1(n)\} \\ X_2(z) &= \mathcal{Z}\{x_2(n)\} \\ k_1, k_2 &\in \mathbb{R} \end{aligned}\right\} \begin{aligned} &\Rightarrow k_1 \cdot X_1(z) + k_2 \cdot X_2(z) \\ &= \mathcal{Z}\{k_1 \cdot x_1(n) + k_2 \cdot x_2(n)\} . \end{aligned} \tag{9.25}$$

2. Delay or advance by N discrete-time units

$$\mathcal{Z}\{x(n)\} = X(z)$$
$$\Rightarrow \begin{cases} \mathcal{Z}\{x(n-N)\} = z^{-N}X(z) \\ \mathcal{Z}\{x(n+N)\} = z^{N}X(z) - \sum\limits_{k=0}^{N-1} z^{N-k}x(k) \end{cases} . \tag{9.26}$$

3. Initial or final value theorems

$$\mathcal{Z}\{x(n)\} = X(z)$$
$$\Rightarrow \begin{cases} x(n=0) = \lim\limits_{z\to\infty}(X(z)) \\ \lim\limits_{n\to\infty}(x(n)) = \lim\limits_{z\to 1}((1-z^{-1})X(z)) \end{cases} . \tag{9.27}$$

Note: the final value theorem ($n \to \infty$) holds only if $X(z)$ is an analytic function of complex variable z.

4. Discrete convolution theorem

$$\left.\begin{aligned} X_1(z) &= \mathcal{Z}\{x_1(n)\} \\ X_2(z) &= \mathcal{Z}\{x_2(n)\} \\ x(n) &= x_1(k) \otimes x_2(k) \triangleq \sum_{k=-\infty}^{+\infty} x_1(k) \cdot x_2(n-k) \end{aligned}\right\}$$
$$\Rightarrow X(z) = \mathcal{Z}\{x(n)\} = X_1(z) \cdot X_2(z) . \tag{9.28}$$

Note: because $x(n)$ needs to be *causal*, i. e., can only depend on values of signals x_1 and x_2 prior or up to instant n, the bounds of the convolutional summation are set as follows,

$$(x_1(k) \otimes x_2(k))(n) = \sum_{k=0}^{n} x_1(k) \cdot x_2(n-k) .$$

It is also pointed out here that the commutative and associative properties hold for the convolution operator applied on two discrete-time signals.

Finally, it is pointed out that the change of variable, $z = \exp(sT_s)$, on the basis of which the Z-transform is obtained from the Laplace transform of signal $x_\delta(t)$, defines a single-valued mapping of complex frequency s-plane to the plane of, also complex, delay operator z. Remember that the stability region on the complex plane $s = a + i\omega$, is the left-hand half-plane defined by inequality $\alpha \leq 0$. In effect, the stability region on the delay operator z-plane is defined by inequality

$$|z| \leq 1 . \tag{9.29}$$

Since

$$z = e^{sT_s} = e^{aT_s}e^{i\omega T_s} \Rightarrow \begin{cases} |z| = e^{aT_s} \\ \arg(z) = e^{i\omega T_s} \end{cases} . \tag{9.30}$$

That is, the stability region is the unit disc. Similarly, the origin of the s-plane is mapped to point $z = A + j0$ on the z-plane, while the imaginary axis of the s-plane is wrapped around the unit circle on the z-plane. Finally, the real s axis is mapped to the real z semi-axis. The Z-transform of several elementary discrete-time signals are shown in Table 9.1. In the interest of completeness and allowing for comparisons a Laplace transform table is provided in Table 9.2.

9.1.5 Discrete-Time LTI Systems

Difference Equations

A special class of dynamic systems, widely used in systems and signal theory, is linear time invariant systems (LTI) [9.1, 2, 4–6]. Such systems are described in continuous time by linear differential equations with constant coefficients. In discrete time, such a differential equation is converted to a difference equation corresponding to a discrete-time system described by a transfer function in variable z, or equivalently z^{-1}. The general form of a linear difference equation with input $u(n)$ and output $y(n)$ is the following,

$$y(n) = \sum_{k=0}^{\mathrm{M}} b_k u(n-k) - \sum_{k=1}^{\mathrm{N}} a_k y(n-k) \,. \tag{9.31}$$

To obtain the iterative process started in order to obtain samples of discrete-time output signal $y(n)$ for $n \geq 0$, N initial conditions are needed: $y(-k)$, $k = 1, \ldots, N$. Using (9.31) we can conclude that the n-th sample of the output is determined as a weighed sum of the current plus M past input samples and N past output samples.

Table 9.1 Some Z-transform pairs

***k*-th term of the time sequence**	***z*-transform**
1 at k, 0 elsewhere (Kronecker delta sequence)	z^{-k}
1 (unit step sequence)	$\dfrac{z}{z-1}$
k (unit ramp sequence)	$\dfrac{z}{(z-1)^2}$
A^k (for complex numbers A)	$\dfrac{z}{z-A}$
kA^k	$\dfrac{Az}{(z-A)^2}$
$\dfrac{(k+1)(K+2)\cdots(K+n-1)}{(n-1)!}A^k$	$\dfrac{z^n}{(z-A)^n}$

Table 9.2 Laplace transform pairs

Time function	**Laplace transform**
Unit impulse $\delta(t)$	1
Unit step $\mathbb{1}(t)$	$\dfrac{1}{s}$
Unit ramp t	$\dfrac{1}{s^2}$
Polynomial t^n	$\dfrac{n!}{s^{n+1}}$
Exponential e^{-at}	$\dfrac{1}{s+a}$
Sine wave $\sin \omega t$	$\dfrac{\omega}{s^2+\omega^2}$
Cosine wave $\cos \omega t$	$\dfrac{s}{s^2+\omega^2}$
Damped sine wave $e^{-at} \sin \omega t$	$\dfrac{\omega}{(s+a)^2+\omega^2}$
Damped cosine wave $e^{-at} \cos \omega t$	$\dfrac{s+a}{(s+a)^2+\omega^2}$

Linear Time-Invariant Discrete-Time Systems

The generic difference (9.31) in the discrete-time domain defines an LTI discrete-time system. In the Z-transform domain the difference equation becomes the transfer function as follows

$$\begin{aligned} H(z^{-1}) &= \frac{Y(z)}{U(z)} \\ &= \frac{b_0 + b_1 z^{-1} + \ldots + b_M z^{-M}}{1 + a_1 z^{-1} + \ldots a_{N-1} z^{-N+1} + a_N z^{-N}} \\ &\Updownarrow \\ H(z) &= \frac{b_0 z^N + b_1 z^{N-1} + \ldots + b_M z^{N-M}}{z^N + a_1 z^{N-1} + \ldots + a_N}\,, \\ M &\leq N \,. \end{aligned} \tag{9.32}$$

A special, yet very important case of discrete-time systems are finite impulse response (FIR) systems in contrast to the generic IIR (infinite impulse response) system defined in (9.31). For a FIR system it holds that $a_{\mathrm{k}} = 0$, $\forall k > 0$ in (9.31), hence

$$\begin{aligned} y(n) &= \sum_{k=0}^{\mathrm{M}} b_{\mathrm{k}} u(n-k) \quad \text{and} \\ H(z^{-1}) &= b_0 + b_1 z^{-1} + \ldots + b_M z^{-M} \,. \end{aligned} \tag{9.33}$$

In effect, the n-th output sample is *exclusively* determined as a weighed sum of the current plus M past input samples. In effect, the difference equations seizes to be recursive and that is why in the literature FIR systems are also mentioned as MA (moving average) ones.

In contrast, discrete-time LTI systems that have exclusively poles, i. e., $b_i = 0$, for every $i > 0$ in (9.32), are also known as AR (auto-regressive), while the generic IIR system is mentioned also as auto-regressive moving average (ARMA) in this framework.

9.1.6 Continuous-Time System Mapping

There are many ways (methods) to map a continuous-time system to a discrete-time equivalent [9.1, 2, 4–6]. All of these methods are based on preserving some characteristic of the continuous-time system during this procedure; this is why there is more than one way. However, *stability* (or instability) of a continuous-time system is preserved by employing any of the methods presented here.

The mapping methods considered here can be classified into the following categories:

1. Approximate differentiation or integration:
   - Backward difference
   - Forward difference
   - Trapezoidal integration (Tustin transformation).

2. Time-domain response matching at sampling instants:
   - Impulse response matching
   - Step response matching.
3. Dynamic matching, i. e., matching of all poles and zeroes appearing in the transfer function.

Approximate Differentiation or Integration

Consider a continuous-time signal $x(t)$ being sampled with sampling interval T_s and the discrete-time signal $x(n)$ is obtained. Then, the following approximations can be derived.

Approximate derivative with respect to time using backward difference

$$\dot{x}(t) \simeq \frac{x(n)-x(n-1)}{T_\mathrm{s}} \Rightarrow s \equiv \frac{1-z^{-1}}{T_\mathrm{s}} . \tag{9.34}$$

Approximate derivative with respect to time using forward difference

$$\dot{x}(t) \simeq \frac{x(n+1)-x(n)}{T_\mathrm{s}} \Rightarrow s \equiv \frac{z-1}{T_\mathrm{s}} = \frac{1-z^{-1}}{T_\mathrm{s} z^{-1}} . \tag{9.35}$$

Approximate integral with respect to time using the trapezoidal rule

$$\begin{aligned} x_1(nT_\mathrm{s}) &= \int\limits_0^{nT_\mathrm{s}} x(\xi)\,\mathrm{d}\xi = \int\limits_0^{(n-1)T_\mathrm{s}} x(\xi)\,\mathrm{d}\xi + \int\limits_{(n-1)T_\mathrm{s}}^{nT_\mathrm{s}} x(\xi)\,\mathrm{d}\xi \\ &\triangleq x_1((n-1)T_\mathrm{s}) + \Delta x_1(n) , \\ \Delta x_1(n) &\simeq T_\mathrm{s} \cdot \frac{x(n)+x(n-1)}{2} . \end{aligned} \tag{9.36}$$

However,

$$\begin{aligned} X_1(s) &= \frac{1}{s} X(s) \quad \text{and} \\ X_1(z) &= z^{-1} X_1(z) + T_\mathrm{s} \frac{1+z^{-1}}{2} X(z) . \end{aligned} \tag{9.37}$$

The above in effect yields

$$\frac{1}{s} \equiv \frac{T_\mathrm{s}}{2} \cdot \frac{1+z^{-1}}{1-z^{-1}} \Leftrightarrow s \equiv \frac{2}{T_\mathrm{s}} \cdot \frac{1-z^{-1}}{1+z^{-1}} . \tag{9.38}$$

Equations (9.34), (9.35), and (9.38) mean that when presented with a transfer function $H(s)$ in the complex frequency (s) domain, describing a continuous-time LTI system, it is possible to derive a discrete-time transfer function $H(z)$, or $H(z^{-1})$, defining a discrete-time LTI system by substituting variable s with the approximation desired.

Impulse Response Matching

The methods presented previously were based on approximating the integration or differentiation operator appearing in the differential (dynamic) equation of an LTI system. However, an LTI system can, also, be described in full by its response to an arbitrary input signal. Common choices include using either Dirac's delta (impulse) or the unit step signal (Heaviside signal).

In effect, if the impulse or step response of a continuous-time system is sampled, then a discrete-time description can be derived as follows.

Invariance of impulse response: the impulse delta is defined as follows in discrete time

$$\delta(n) = \begin{cases} 0, & n \neq 0 \\ 1, & n = 0 \end{cases} \Leftrightarrow \Delta(z) = 1 . \tag{9.39}$$

Given the transfer function, $H(z)$ of a discrete-time LTI system, it is easy to prove that, just like in the continuous time case, $H(z)$ is the Z-transform of the system's impulse response.

Indeed, the output $y(n)$ of a discrete-time LTI system is given by the following convolutional sum

$$\begin{aligned} y(n) = u(k) \otimes h(k) &\triangleq \sum_{k=-\infty}^{+\infty} u(k) \cdot h(n-k) \Rightarrow Y(z) \\ &= H(z) \cdot U(z) . \end{aligned} \tag{9.40}$$

In effect,

$$u(n) = \delta(n) \Rightarrow U(z) = 1 \Rightarrow Y(z) = H(z) .$$

Consider a continuous-time LTI system with impulse response $h(t) = \mathcal{L}^{-1}\{H(s)\}$, which is being sampled at rate f_s. This means that continuous-time signal $h_\delta(t) = \sum_{n=-\infty}^{+\infty} h(nT_\mathrm{s})\delta(t-nT_\mathrm{s})$ corresponds to discrete-time signal $h_\delta(n) \equiv h(nT_\mathrm{s})$, i. e., a sample set of $h(t)$.

The Z-transform of $h_\delta(n)$,

$$H_\delta(z^{-1}) = \sum_{n=0}^{+\infty} h_\delta(n) z^{-n} ,$$

defines a discrete-time LTI system, the impulse response of which, by definition, coincides with that of continuous-time transfer function $H(s)$, at all sampling instants.

Assume, now, that for every positive M there is a positive integer ε such that

$$\sum_{n=M+1}^{+\infty} h_\delta^2(n) < \varepsilon . \tag{9.41}$$

Given the above, one can proceed to impulse response truncation with accuracy ε. This procedure consists of approximating the original IIR system with transfer function $H_8(z^{-1}) = \sum_{n=0}^{+\infty} h_8(n)z^{-n}$ by the following FIR one with memory M

$$\hat{H}_{\mathrm{FIR}}(z^{-1}) = \sum_{n=0}^{M} h_8(n)z^{-n} = \sum_{n=0}^{M} b_n z^{-n} \,. \tag{9.42}$$

Approximating a continuous-time system with a discrete-time one, which on top might be FIR, comes with a number of advantages, especially for real-time applications.

Invariance of step response: the unit step signal in discrete time is defined as follows

$$u_{\mathrm{step}}(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \Leftrightarrow$$

$$U_{\mathrm{step}}(z) = \sum_{n=0}^{+\infty} z^{-n} = \frac{1}{1-z^{-1}} \,. \tag{9.43}$$

The Z-transform of the unit step signal is derived by applying the lemma for the infinite sum of a geometric progression with a ratio that in magnitude is less than 1.

Consider now an LTI system with impulse response as follows

$$y_{\mathrm{step}}(t) = \mathcal{L}^{-1}\{H(s)U_{\mathrm{step}}(s)\} = \mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\} \,. \tag{9.44}$$

We will now determine a discrete-time transfer function $H(z)$ such that the latter's unit step response coincides with that of the continuous-time system above. Mathematically, this is defined as follows

$$\mathcal{Z}^{-1}\{H(z)U_{\mathrm{step}}(z)\} = y_{\mathrm{step}}(n) = y_{\mathrm{step}}(nT_{\mathrm{s}}) \,. \tag{9.45}$$

Then, by employing (9.43), one obtains that

$$H(z) = (1-z^{-1})\mathcal{Z}\{y_{\mathrm{step}}(nT_{\mathrm{s}})\} \,. \tag{9.46}$$

Since

$$z^{-1} = \exp(-sT_{\mathrm{s}}) \quad \text{and} \quad Y_{\mathrm{step}}(s) = \frac{H(s)}{s} \,.$$

We can finally obtain that

$$H(z) = \mathcal{Z}\{y_{\mathrm{ZOH}}(nT_{\mathrm{s}})\} \,. \tag{9.47}$$

In the above, it holds that

$$y_{\mathrm{ZOH}}(t) = h_{\mathrm{ZOH}}(t) \otimes h(t) \Leftrightarrow$$

$$Y_{\mathrm{ZOH}}(s) = H_{\mathrm{ZOH}}(s)H(s) = \frac{1-e^{-sT_{\mathrm{s}}}}{s}H(s) \,. \tag{9.48}$$

Pole and Zero Matching of a Transfer Function

This method derives from equation

$$z^{-1} = \exp(-sT_{\mathrm{s}})$$

connecting variable z with complex frequency s. Combining the above with the following factorized form of a continuous-time scalar transfer function

$$H(s) = K_{\mathrm{s}} \frac{(s+z_1)(s+z_2)\ldots(s+z_m)}{(s+p_1)(s+p_2)\ldots(s+p_n)}, \quad m \leq n \,. \tag{9.49}$$

One can map every pole and every zero of the continuous-time transfer function to a corresponding pole and zero of an equivalent discrete-time transfer function

$$\begin{aligned} \zeta_{\mathrm{i}} &= -\exp(-T_{\mathrm{s}}z_{\mathrm{i}}), \quad i = 1, \ldots, m \,, \\ \pi_{\mathrm{i}} &= -\exp(-T_{\mathrm{s}}p_{\mathrm{i}}), \quad i = 1, \ldots, n \,. \end{aligned} \tag{9.50}$$

In effect, the following discrete-time transfer function is derived in its factorized from

$$H(z) = k_z \frac{(z+1)^{n-m}(z+\zeta_1)(z+\zeta_2)\ldots(z+\zeta_m)}{(z+\pi_1)(z+\pi_2)\ldots(z+\pi_n)} \,. \tag{9.51}$$

Factor $(z+1)^{n-m}$ is introduced in the discrete-time transfer function above in correspondence to factor $(\tau s+1)^{n-m}$ with $\tau \to \pm j0$, which is *hidden* in the continuous-time transfer function (9.49). The intent is to equate the numerator degree to that of the denominator. This is achieved by introducing an extra zero at imaginary infinity ($\pm j\infty$) with multiplicity $(n-m)$ using factor $(\tau s+1)^{n-m}$. In the discrete-time transfer function, factor $(\tau s+1)^{n-m}$ is mapped to $(z+1)^{n-m}$, since point $\pm j\infty$ on the s-plane is mapped to -1 on the z-plane.

Finally, gain k in the discrete-time transfer function is determined by equating its value at a certain z with that of the continuous-time transfer function at the equivalent point s. For example, for $z = 1$, the corresponding point is $s = 0$, and, therefore, k is calculated as follows

$$\begin{aligned} H(z)|_{z=1} &= H(s)|_{s=0} \\ &\Leftrightarrow k_z \frac{2^{n-m}(1+\zeta_1)(1+\zeta_2)\ldots(1+\zeta_m)}{(1+\pi_1)(1+\pi_2)\ldots(1+\pi_n)} \,, \\ &= K_{\mathrm{s}} \frac{z_1 z_2 \ldots z_m}{p_1 p_2 \ldots p_n} \,. \end{aligned} \tag{9.52}$$

9.2 Digital Filters

In the previous section we studied different ways of describing discrete-time systems that are linear and time invariant. It was verified that the Z-transform greatly simplifies the analysis of discrete-time systems, especially those initially described by difference equations.

In this section, we study in finer detail several structures used to realize a given transfer function associated with a specific difference equation through the use of the Z-transform. The transfer functions considered here will be of the polynomial form (non-recursive filters, FIR) and of the rational-polynomial form (recursive filters, IIR). In the non-recursive case we emphasize the existence of the important subclass of linear-phase filters. Then we introduce some tools to calculate the digital network transfer function, as well as to analyze its internal behavior. We also discuss some properties of generic digital filter structures associated with practical discrete-time systems.

9.2.1 Important FIR Filter Structures

Non-recursive filters are characterized by a difference equation in the form

$$y(n) = \sum_{l=0}^{M} b_1 x(n-1) , \tag{9.53}$$

where the b_l coefficients are directly related to the system impulse response; that is, $b_1 = h(l)$. Owing to the finite length of their impulse responses, non-recursive filters are also referred to as finite-duration impulse response (FIR) filters. We can rewrite (9.53) as follows

$$y(n) = \sum_{l=0}^{\mathrm{M}} h(l)x(n-l) . \tag{9.54}$$

Applying the Z-transform to the equation above, we end up with the following input–output relationship

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{l=0}^{M} h(l)z^{-l} = \sum_{l=0}^{M} b_l z^{-l} . \tag{9.55}$$

In practical terms, (9.55) can be implemented in several distinct forms, using as basic elements the delay, the multiplier, and the adder blocks. These basic elements of digital filters and their corresponding standard symbols are depicted in Fig. 9.8. An alternative way of representing such elements is the so-called signal flow graph shown in Fig. 9.9.

These two sets of symbolisms representing the delay, multiplier, and adder elements are used throughout this text interchangeably.

Direct Form

The simplest realization of an FIR digital filter is derived from (9.55). The resulting structure, which can be seen in Fig. 9.10, is called the direct-form realization, as the multiplier coefficients are obtained directly from the filter transfer function. Such a structure is also referred to as the canonic direct form, where we understand canonic form to mean any structure that realizes a given transfer function with the minimum number of delays,

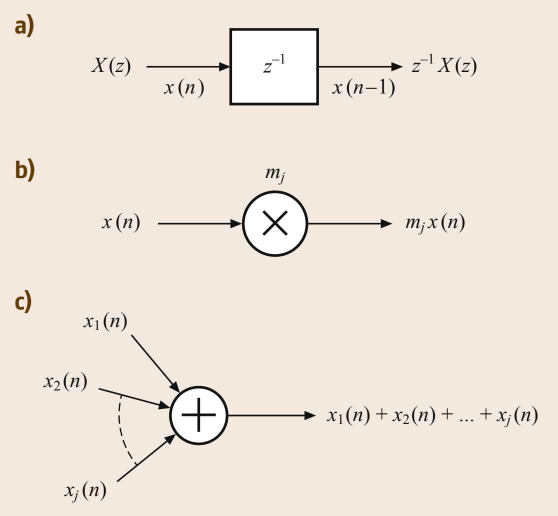


Fig. 9.8a–c Classic representation of basic elements of digital filters: **(a)** delay; **(b)** multiplier; **(c)** adder

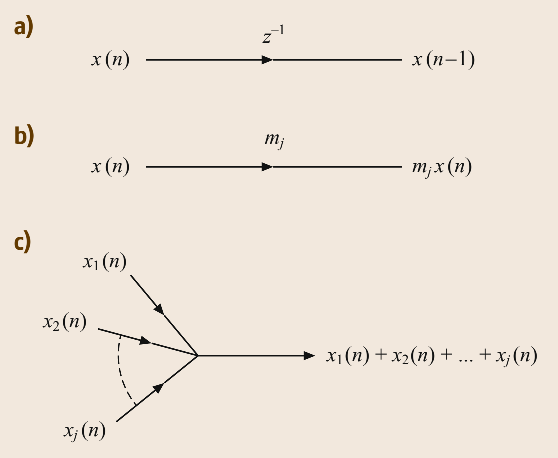


Fig. 9.9a–c Signal-flow graph representation of basic elements of digital filters: **(a)** delay; **(b)** multiplier; **(c)** adder

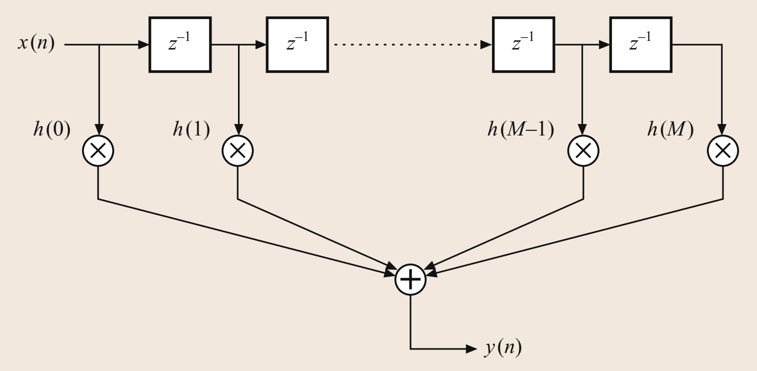


Fig. 9.10 Direct form for FIR digital filters

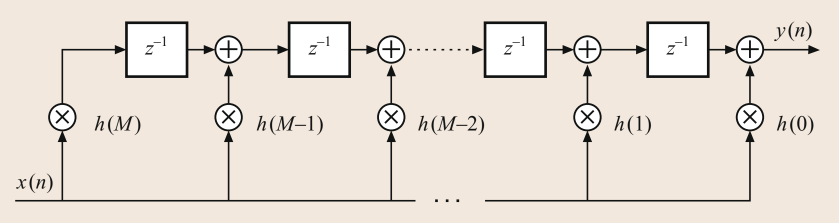


Fig. 9.11 Alternative direct form for FIR digital filters

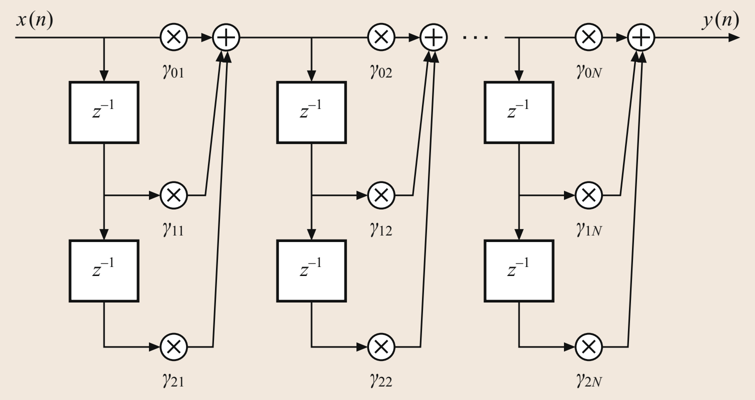


Fig. 9.12 Cascade form for FIR digital filters

multipliers, and adders. More specifically, a structure that utilizes the minimum number of delays is said to be canonical with respect to the delay element, and so on.

An alternative canonical direct form for (9.76) can be derived by expressing $H(z)$ as follows

$$(z)=\sum_{l=0}^{M} h(l)z^{-l} = h(0)+z^{-1}\left(\sum_{l=1}^{M} h(l)z^{-l}\right) . \qquad (9.56)$$

The implementation of this form is shown in Fig. 9.11.

Cascade Form

Equation (9.55) can be realized through a series of equivalent structures. However, the coefficients of such distinct realizations may not be explicitly the filter impulse response or the corresponding transfer function. An important example of such a realization is the so-called cascade form, which consists of a series of second-order FIR filters connected in cascade, thus the name of the resulting structure, as seen in Fig. 9.12. The transfer function associated with such a realization is of the form

$$H(z)=\prod_{k=1}^{N}\left(\gamma_{0k}+\gamma_{1k}z^{-1}+\gamma_{2k}z^{-2}\right) . \qquad (9.57)$$

In the above, if M is the filter order, then $N = M/2$ when M is even and $N = (M+1)/2$ when M is odd. In the latter case, one of the γ_{2k} vanishes.

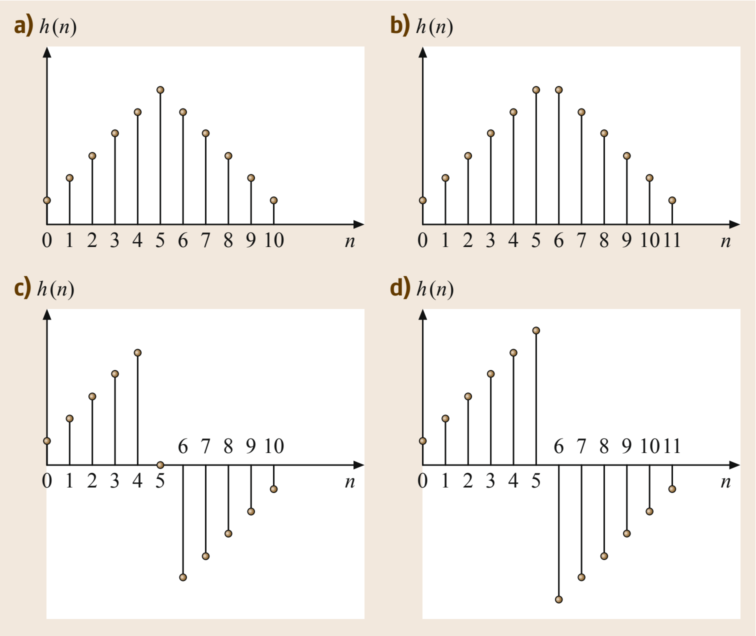


Fig. 9.13a–d Example of impulse responses of linear-phase FIR digital filters: **(a)** type I; **(b)** type II; **(c)** type III; **(d)** type IV

Linear-Phase Form

An important subclass of FIR digital filters is the one that includes linear-phase filters. Such filters are characterized by a constant group delay τ; therefore, they must present a frequency response of the following form

$$H(e^{i\omega}) = B(\omega)\exp(-i\omega\tau + i\varphi)\ . \tag{9.58}$$

In the above $B(\omega)$ is real and τ and φ are constant. We now proceed to show that linear-phase FIR filters present impulse responses of very particular forms. Specifically, if $h(n)$ is to be causal and of finite duration, for $0 \le n \le M$, we must necessarily have that

$$\tau = \frac{M}{2}\ . \tag{9.59}$$

Therefore, one obtains that

$$h(n) = e^{2i\varphi} h^*(M-n)\ . \tag{9.60}$$

This is the general equation that the coefficients of a linear-phase FIR filter must satisfy. In the common case, where all the filter coefficients are real, one finally obtains that the filter impulse response must be either *symmetric* or *antisymmetric*. In effect, the frequency response of linear-phase FIR filters with real coefficients becomes as follows

$$H(e^{i\omega}) = B(\omega)\exp\left(-i\omega\left(\frac{M}{2}\right) + i\left(\frac{k\pi}{2}\right)\right). \tag{9.61}$$

As a result, solely four distinct cases need be considered that are described by the equations above. Their types have been standardized and are referred to in literature as follows [9.2]:

- Type I: $k = 0$ and M even
- Type II: $k = 0$ and M odd
- Type III: $k = 1$ and M even
- Type IV: $k = 1$ and M odd.

Typical impulse responses of the four cases of linear-phase FIR digital filters are depicted in Fig. 9.13.

9.2.2 Important IIR Filter Structures

Direct Form

Recursive filters have transfer functions of the following form

$$H(z) = \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^{M} b_i z^{-i}}{1 + \sum_{i=1}^{N} a_i z^{-i}}\ . \tag{9.62}$$

Since, in most cases, such transfer functions give rise to filters with impulse responses having infinite durations, recursive filters are also referred to as infinite-duration impulse response (IIR) filters.

We can consider that $H(z)$ as above results from the cascading of two separate filters of transfer functions $N(z)$ and $1/D(z)$. The $N(z)$ polynomial can be realized with the FIR direct form, as shown in the previous section. The realization of $1/D(z)$ can be performed as depicted in Fig. 9.14, where the FIR filter shown will

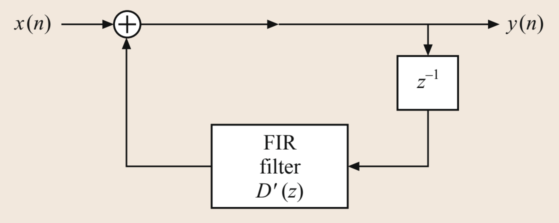


Fig. 9.14 Block diagram realization of $1/D(z)$

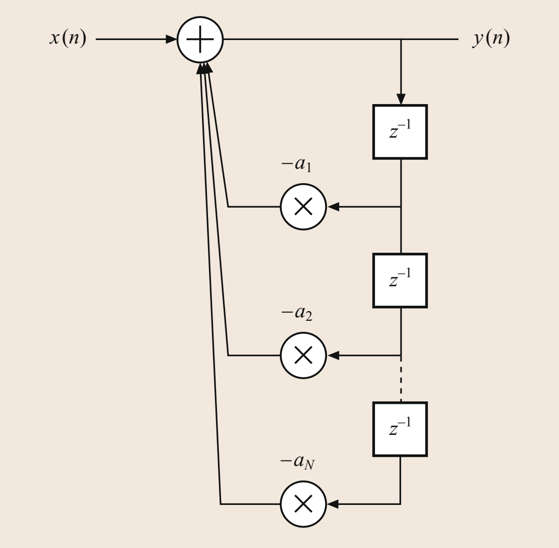


Fig. 9.15 Detailed realization of $1/D(z)$

be an $(N-1)$-th order filter with transfer function as follows

$$D'(z) = z(1-D(z)) = -z\sum_{i=1}^{N} a_i z^{-i} \,. \tag{9.63}$$

The direct form of realizing $1/D(z)$ is shown in Fig. 9.15.

The complete realization of $H(z)$, as a cascade of $N(z)$ and $1/D(z)$, is shown in Fig. 9.16. Such a structure is not canonic with respect to the delays, since for an (M, N)-th order filter this realization requires $(N+M)$ delays.

Clearly, in the general case we can change the order in which we cascade the two separate filters; that is, $H(z)$ can be realized as

$$N(z) \times \frac{1}{D(z)} \quad \text{or} \quad \left(\frac{1}{D(z)}\right) \times N(z) \,.$$

In the second option, all delays employed start from the same node, which allows us to eliminate the consequent redundant delays. In that manner, the resulting structure, usually referred to as the Type 1 canonic direct form, is the one depicted in Fig. 9.17, for the special case $N = M$.

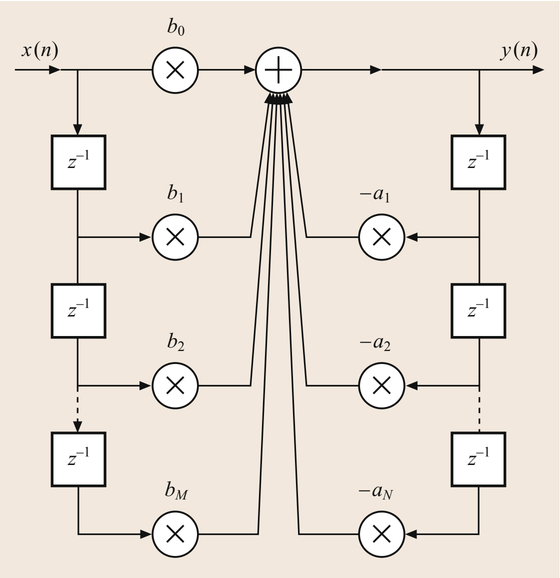


Fig. 9.16 Non-canonic IIR direct form realization

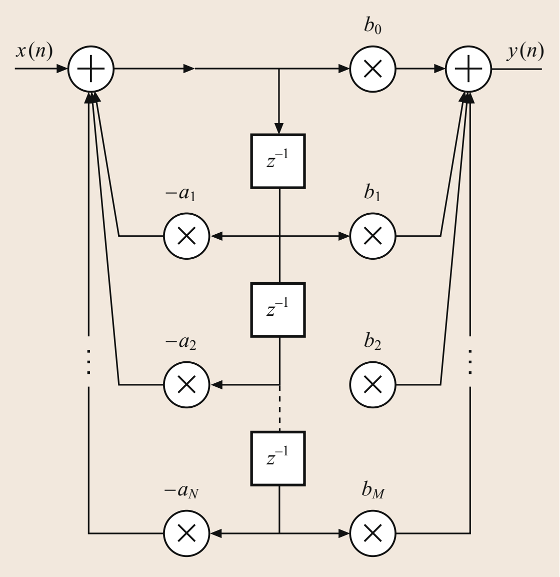


Fig. 9.17 Type 1 canonic direct form for IIR filters

An alternative structure, the so-called Type 2 canonic direct form, is shown in Fig. 9.18. Such a realization is generated from the corresponding nonrecursive form.

The majority of IIR filter transfer functions used in practice present a numerator degree M smaller than or equal to the denominator degree N. In general, one can consider, without much loss of generality, that $M = N$.

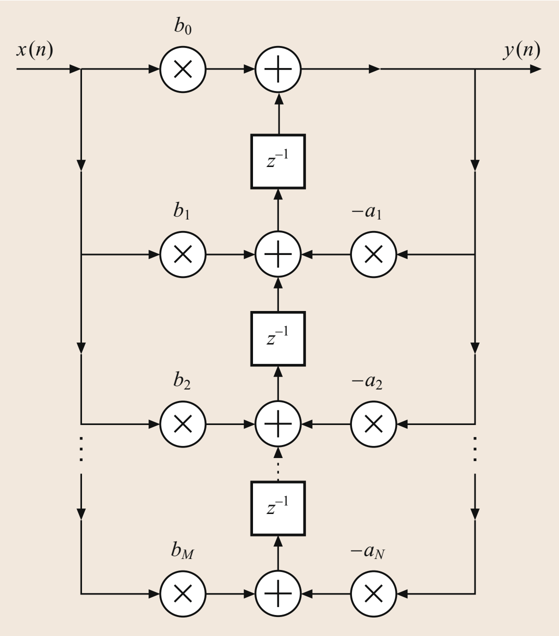


Fig. 9.18 Type 2 canonic direct form for IIR filters

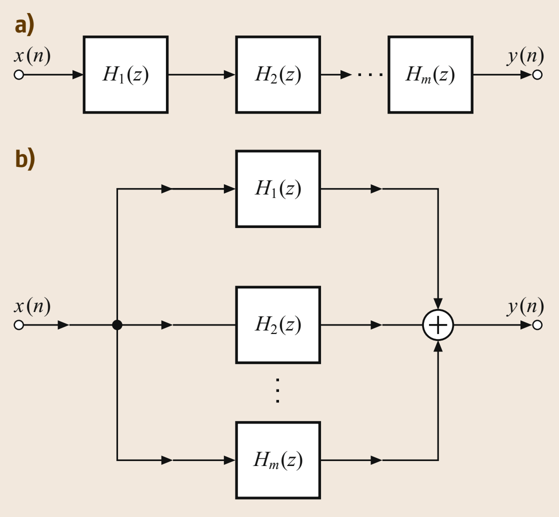


Fig. 9.19a,b Block diagrams of: **(a)** cascade form; **(b)** parallel form

In the case where $M < N$, we just make the coefficients $b_{M+1}, b_{M+2}, \dots, b_N$ in Figs. 9.17 and 9.18 equal to zero.

Cascade Form

In the same way as their FIR counterparts, IIR digital filters present a large variety of possible alternative realizations. An important one, referred to as cascade realization, is depicted in Fig. 9.19a, where the basic blocks represent simple transfer functions of orders 2 or 1. In fact, the cascade form, based on second-order blocks, is associated with the following transfer function decomposition

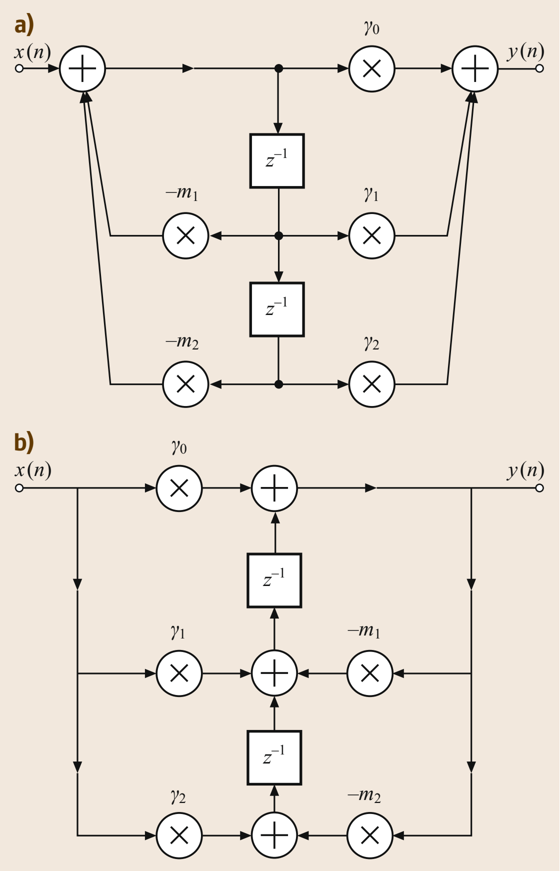


Fig. 9.20a,b Block diagrams of: **(a)** cascade form; **(b)** parallel form

$$\begin{aligned} H(z) &= \prod_{k=1}^{m} \frac{\gamma_{0k} + \gamma_{1k} z^{-1} + \gamma_{2k} z^{-2}}{1 + m_{1k} z^{-1} + m_{2k} z^{-2}} , \\ &= \prod_{k=1}^{m} \frac{\gamma_{0k} z^2 + \gamma_{1k} z + \gamma_{2k}}{z^2 + m_{1k} z + m_{2k}} , \\ &= H_0 \prod_{k=1}^{m} \frac{z^2 + \gamma'_{1k} z + \gamma'_{2k}}{z^2 + m_{1k} z + m_{2k}} . \end{aligned} \tag{9.64}$$

Parallel Form

Another important realization for recursive digital filters is the parallel form represented in Fig. 9.19b. Using second-order blocks, which are the most commonly used in practice, the parallel realization corresponds to the following transfer function decomposition

$$
\begin{aligned}
H(z) &= \prod_{k=1}^{m} \frac{\gamma_{0k}^{\mathrm{p}} z^2 + \gamma_{1k}^{\mathrm{p}} z + \gamma_{2k}^{\mathrm{p}}}{z^2 + m_{1k} z + m_{2k}} \,, \\
&= h_0 + \prod_{k=1}^{m} \frac{\gamma_{1k}^{\mathrm{p}'} z + \gamma_{2k}^{\mathrm{p}'}}{z^2 + m_{1k} z + m_{2k}} \,, \\
&= h_0' + \prod_{k=1}^{m} \frac{\gamma_{0k}^{\mathrm{p}''} z^2 + \gamma_{1k}^{\mathrm{p}''} z}{z^2 + m_{1k} z + m_{2k}} \,. \qquad (9.65)
\end{aligned}
$$

The above is also known as the partial-fraction decomposition. This equation indicates three alternative forms of the parallel realization, where the last two are canonic with respect to the number of multiplier elements.

It should be mentioned that each second-order block in the cascade and parallel forms can be realized by any of the existing distinct structures, as, for instance, one of the direct forms shown in Fig. 9.20.

All these digital filter realizations present different properties when one considers practical finite-precision implementations; that is, the quantization of the coefficients and the finite precision of the arithmetic operations, such as additions and multiplications. In fact, the analysis of the finite-precision effects in the distinct realizations is a fundamental step in the overall process of designing any digital filter [9.2].

9.3 The Fast Fourier Transform (FFT)

The FFT (fast Fourier transform) algorithm is a faster version of the discrete Fourier transform (DFT). FFT utilizes some clever algorithms to do the same thing as the DFT, but in much less time [9.2].

The DFT is extremely important in the area of frequency (spectral) analysis because it takes a discrete signal in the time domain and transforms that signal into its discrete frequency domain representation. Without a discrete-time to discrete-frequency transform we would not be able to compute the Fourier transform with a microprocessor or DSP-based (DSP: Digital Signal Processor) system [9.1, 2, 4–6].

It is the speed and discrete nature of the FFT that allows us to analyze a signal's spectrum, as will soon become evident.

9.3.1 Review of Integral Transforms

We first give a review of the integral transforms that have been used in the text, possibly in their unilateral (single-side) version.

The bilateral Laplace transform

$$
X(s) = \mathcal{L}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) e^{-st} \, \mathrm{d}t, x(t) \overset{\mathcal{L}}{\rightleftarrows} X(s) \,. \qquad (9.66)
$$

The continuous-time Fourier transform

$$
X(i\omega) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) e^{-i\omega t} \, \mathrm{d}t, x(t) \overset{\mathcal{F}}{\rightleftarrows} X(i\omega) \,. \qquad (9.67)
$$

The bilateral Z-transform

$$
\begin{aligned}
X(z) &= \mathcal{Z}\{x(n)\} \\
&= \sum_{n=-\infty}^{+\infty} x(n) z^{-n}, x(n) \overset{\mathcal{Z}}{\rightleftarrows} X(z) \,. \qquad (9.68)
\end{aligned}
$$

The Laplace transform is used to obtain a pole-zero representation of a continuous-time signal or system, $x(t)$, in the s-plane. Similarly, the Z-transform is used to find a pole-zero representation of a discrete-time signal or system, $x(n)$, in the z-plane.

The continuous-time Fourier transform can be found by evaluating the Laplace transform form at $s = i\omega$. The picture can be extended by introducing the discrete-time Fourier transform (DTFT). The DTFT can be found by evaluating the Z-transform at $z = \exp(i\Omega)$, as follows

$$
X(e^{i\Omega}) = \sum_{n=-\infty}^{+\infty} x(n) e^{-i\Omega n}, x(n) \overset{\text{DTFT}}{\rightleftarrows} X(e^{i\Omega}) \,. \qquad (9.69)
$$

One needs to point out here that the frequency variable Ω is in normalized units of radians per sample rather than *absolute* units of rad/s, which apply to ω appearing in the standard Fourier transform. This can be justified by recollecting that sequences $x(n)$ are generated by sampling a continuous-time signal $x(t)$ at a certain sampling rate f_s. In this respect, the DTFT can be viewed as a discrete approximation of the Fourier

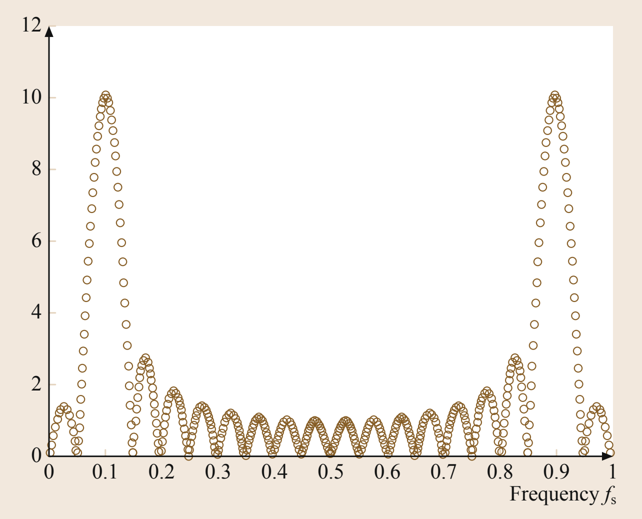


Fig. 9.21 Plot showing the symmetry of DFT

transform as explained below

$$\mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t)e^{-i\omega t}\,\mathrm{d}t$$
$$\simeq T_s \sum_{n=-\infty}^{+\infty} x(nT_s)\exp(-i\omega nT_s)$$
$$\Downarrow$$
$$\mathcal{F}\{x(t)\} \simeq T_s \cdot X(e^{i\Omega}),\, \Omega = \omega T_s = 2\pi \frac{f}{f_s}\,. \tag{9.70}$$

9.3.2 The Discrete Fourier Transform (DFT)

First of all, the discrete Fourier transform (DFT) is *not* the same as the DTFT. Both start with a discrete-time signal, but DFT produces a discrete frequency domain representation while DTFT is continuous in the frequency domain. These two transforms have much in common, however. It is, therefore, helpful to have a basic understanding of the properties of the DTFT.

Periodicity

DTFT is periodic because of the fact that the signal is a discrete-time one. Indeed,

$$X(e^{i(\Omega+2\pi k)}) = \sum_{n=-\infty}^{+\infty} x(n)e^{-i(\Omega+2\pi k)n}$$
$$= \sum_{n=-\infty}^{+\infty} x(n)e^{-i\Omega n}e^{-i2\pi kn} = X(e^{i\Omega})\,. \tag{9.71}$$

One fundamental period is, therefore, 2π, i. e., extends from $f = 0$ to f_s, where f_s is the sampling frequency. Taking advantage of this redundancy, the DFT is only defined in the region between 0 and f_s, in terms of f, or, 0 and 2π, in terms of Ω.

Symmetry

When the region between 0 and f_s is examined, it can be seen that there is even symmetry around the center point, i. e., point $f_s/2$ (half the sampling rate). Indeed,

$$\begin{aligned}
&\Omega_{1,2} = \pi \pm \varepsilon, 0 < \varepsilon < \pi :\\
&X(e^{i\Omega_2})\\
&= \sum_{n=-\infty}^{+\infty} x(n)e^{-i(\pi+\varepsilon)n} = \sum_{n=-\infty}^{+\infty} x(n)e^{-i\varepsilon n}e^{-i\pi n}\\
&= \sum_{n=-\infty}^{+\infty} x(n)e^{-i\varepsilon n}e^{+i\pi n} = \sum_{n=-\infty}^{+\infty} x(n)e^{-i(\pi-\varepsilon)n}\\
&= X(e^{i\Omega_1})\,.
\end{aligned} \tag{9.72}$$

This symmetry adds redundant information. Figure 9.21 shows the DFT (implemented with Matlab's FFT function) of a cosine with a frequency one tenth of the sampling frequency. Note that the data between $0.5f_s$ and f_s is a mirror image of the data between 0 and $0.5f_s$.

Therefore, the discrete Fourier transform (DFT) can be introduced as follows

$$\begin{aligned}
&X_k = \sum_{n=0}^{N-1} x_n \exp\left(-\frac{i2\pi nk}{N}\right),\\
&k = 0, 1, 2, \ldots, (N-1)\,.
\end{aligned} \tag{9.73}$$

Note that the above is actually a transformation between a finite-length real or complex sequence x_n, corresponding to a sampled, at rate $f_s = 1/T_s$, segment of a real or complex signal with actual duration NT_s, to a generally complex sequence (discrete spectrum) of equal finite length N corresponding to frequency values in the range between 0 and $(N-1)f_s/N$ with a step equal to f_s/N; note that the corresponding range in terms of Ω is 0 to $(N-1)2\pi/N$ and the step $2\pi/N$.

Fast Fourier Transform (FFT)

FFT is simply an algorithm to speed up the DFT calculation by reducing the number of multiplications and additions required. It was popularized by J.W. Cooley and J.W. Tukey in the 1960s and, was actually a rediscovery of an idea of Runge (1903) and Danielson and Lanczos (1942), first occurring prior to the availability of computers and calculators – when numerical calculation could take many man hours. In ad-

Table 9.3 8-point DFT ($N = 8$)

$X(k) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{\frac{-i2\pi nk}{N}} = \frac{1}{N}\sum_{n=0}^{N-1} x(n){W_N}^{nk} \quad W_N = e^{\frac{-i2\pi}{N}}$
$X(0) = x(0)W_8^0 + x(1)W_8^0 + x(2)W_8^0 + x(3)W_8^0 + x(4)W_8^0 + x(5)W_8^0 + x(6)W_8^0 + x(7)W_8^0$
$X(1) = x(0)W_8^0 + x(1)W_8^1 + x(2)W_8^2 + x(3)W_8^3 + x(4)W_8^4 + x(5)W_8^5 + x(6)W_8^6 + x(7)W_8^7$
$X(2) = x(0)W_8^0 + x(1)W_8^2 + x(2)W_8^4 + x(3)W_8^6 + x(4)W_8^8 + x(5)W_8^{10} + x(6)W_8^{12} + x(7)W_8^{14}$
$X(3) = x(0)W_8^0 + x(1)W_8^3 + x(2)W_8^6 + x(3)W_8^9 + x(4)W_8^{12} + x(5)W_8^{15} + x(6)W_8^{18} + x(7)W_8^{21}$
$X(4) = x(0)W_8^0 + x(1)W_8^4 + x(2)W_8^8 + x(3)W_8^{12} + x(4)W_8^{16} + x(5)W_8^{20} + x(6)W_8^{24} + x(7)W_8^{28}$
$X(5) = x(0)W_8^0 + x(1)W_8^5 + x(2)W_8^{10} + x(3)W_8^{15} + x(4)W_8^{20} + x(5)W_8^{25} + x(6)W_8^{30} + x(7)W_8^{35}$
$X(6) = x(0)W_8^0 + x(1)W_8^6 + x(2)W_8^{12} + x(3)W_8^{18} + x(4)W_8^{24} + x(5)W_8^{30} + x(6)W_8^{36} + x(7)W_8^{42}$
$X(7) = x(0)W_8^0 + x(1)W_8^7 + x(2)W_8^{14} + x(3)W_8^{21} + x(4)W_8^{28} + x(5)W_8^{35} + x(6)W_8^{42} + x(7)W_8^{49}$
N^2 Complex multiplications
$\frac{1}{N}$ Scaling factor omitted

Table 9.4 8-point DFT. Applying the properties of symmetry and periodicity to W_N^r for $N = 8$

$N = 8$	$W_8^4 = W_8^{0+4} = -W_8^0 = -1$
	$W_8^5 = W_8^{1+4} = -W_8^1$
	$W_8^6 = W_8^{2+4} = -W_8^2$
	$W_8^7 = W_8^{3+4} = -W_8^3$
	$W_8^8 = W_8^{0+8} = +W_8^0 = +1$
	$W_8^9 = W_8^{1+8} = +W_8^1$
	$W_8^{10} = W_8^{2+8} = +W_8^2$
	$W_8^{11} = W_8^{3+4} = +W_8^3$
	• • •
	• • •
	• • •
Symmetry: $W_N^{r+N/2} = -W_N^r$, Periodicity: $W_N^{r+N} = W_N^r$	

Table 9.5 FFT versus DFT. FFT is simply an algorithm for efficiently calculating DFT. Computational efficiency of an N-point FFT: 1) DFT: N^2 Complex multiplications; 2) FFT: $(N/2)\log_2(N)$ Complex multiplications

N	DFT multiplications	FFT multiplications	FFT efficiency
256	65 536	1024	64 : 1
512	262 144	2304	114 : 1
1024	1 048 576	5120	205 : 1
2048	4 194 304	11 264	372 : 1
4096	16 777 216	24 576	683 : 1

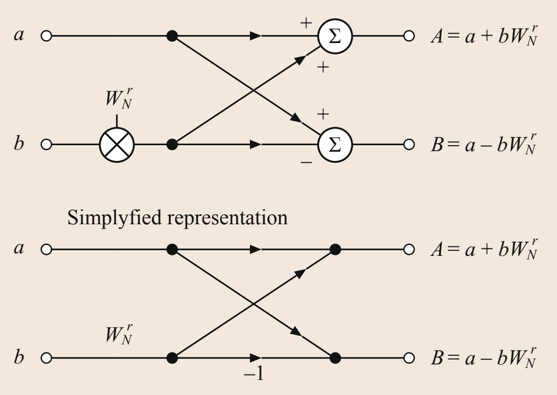


Fig. 9.22 The basic butterfly computation in the DIT-FFT algorithm

dition, the German mathematician Carl Friedrich Gauss (1777–1855) had used the method more than a century earlier.

In order to understand the basic concepts of FFT and its derivation, note that the DFT expansion shown in Table 9.3 can be greatly simplified by taking advantage of the symmetry and periodicity of the twiddle factors as shown in Table 9.4. If the equations are rearranged and factored, the result is the fast Fourier transform (FFT), which requires only $(N/2)\log_2(N)$ complex multiplications. The computational efficiency of FFT versus DFT becomes highly significant when the FFT point size increases to several thousand, as shown in Table 9.5. However, notice that FFT computes *all* the output frequency components (either all or none!). If only a few spectral points need to be calculated, DFT may actually be more efficient. Calculation of a single spectral output using the DFT requires only N complex multiplications.

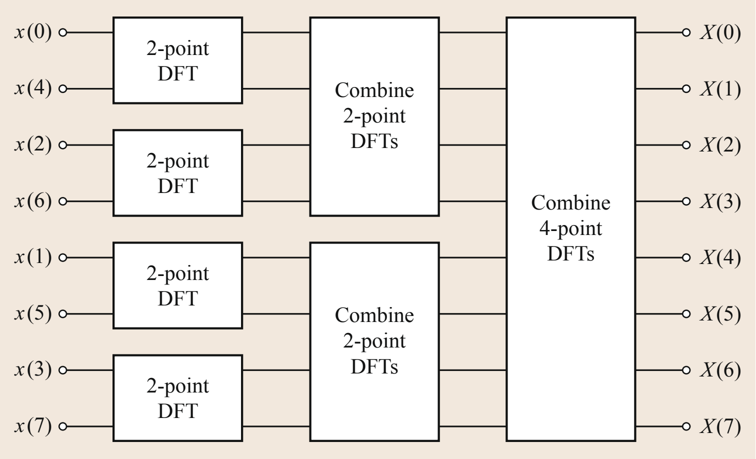


Fig. 9.23 Computation of eight-point FFT in three stages using DIT

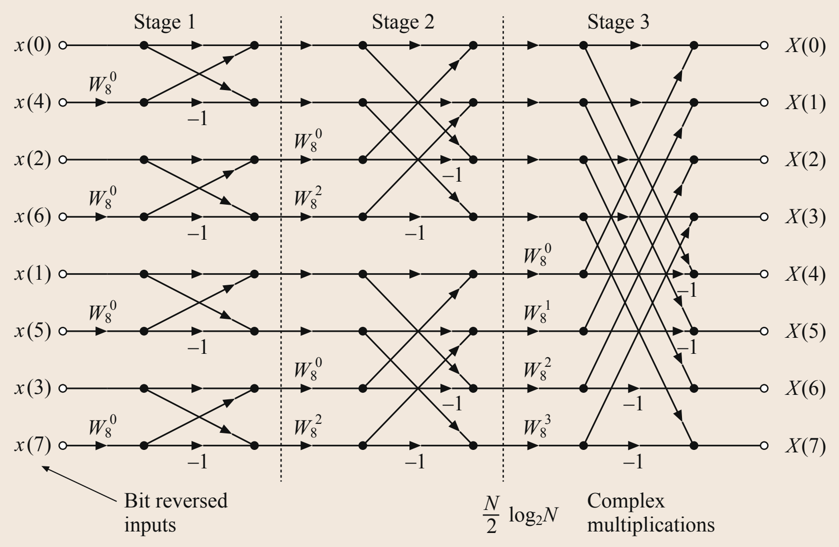


Fig. 9.24 Eight-point DIT-FFT algorithm

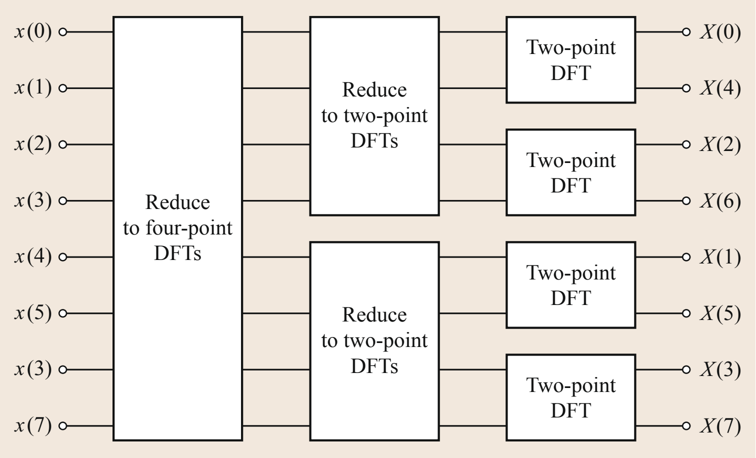


Fig. 9.25 Computation of eight-point FFT in three stages using DIF

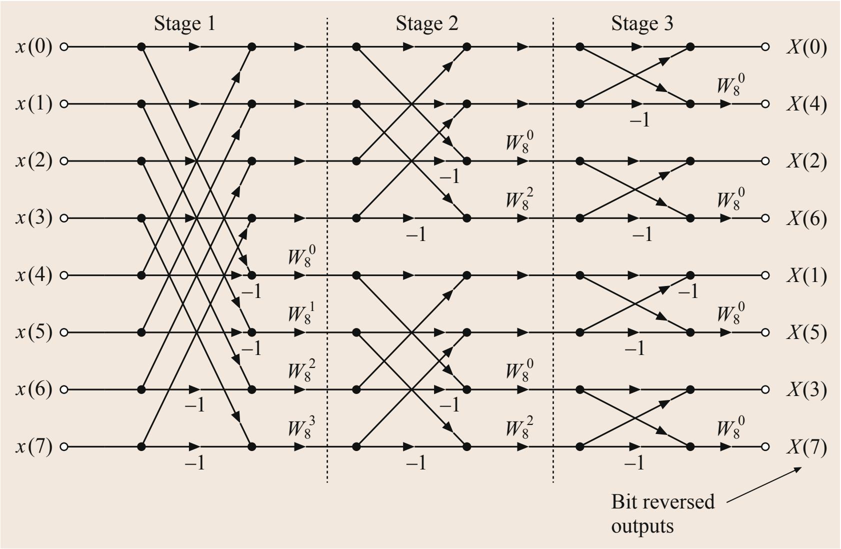


Fig. 9.26 Eight-point DIF-FFT algorithm

The radix-2 FFT algorithm breaks the entire DFT calculation down into a number of two-point DFTs. Each two-point DFT consists of a multiply-and-accumulate operation called a *butterfly*, as shown in Fig. 9.22. Two representations of the butterfly are shown in the diagram: the top diagram is the actual functional representation of the butterfly showing the digital multipliers and adders. In the simplified bottom diagram, the multiplications are indicated by placing the multiplier over an arrow, and addition is indicated whenever two arrows converge at a dot.

The eight-point decimation-in-time (DIT) FFT algorithm computes the final output in three stages as shown in Fig. 9.23. The eight input time samples are first divided (or *decimated*) into four groups of two-point DFTs. The four two-point DFTs are then combined into two four-point DFTs. The two four-point DFTs are then combined to produce the final output $X(k)$. The detailed process is shown in Fig. 9.24, where all the multiplications and additions are shown. Note that the basic two-point DFT butterfly operation forms the basis for all computations. The computation is done in three stages. After the first stage computation is complete, there is no need to store any previous results. The first stage outputs can be stored in the same registers that originally held the time samples $x(n)$. Similarly, when the second stage computation is completed, the results of the first stage computation can be deleted.

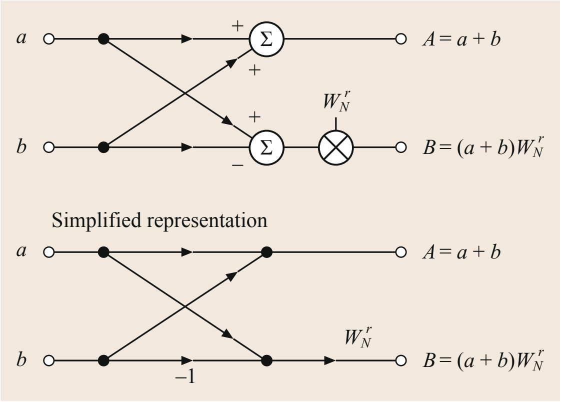


Fig. 9.27 The basic butterfly computation in the DIF-FFT algorithm

In this way, *in-place* computation proceeds to the final stage. Note that in order for the algorithm to work properly, the order of the input time samples, $x(n)$, must be properly re-ordered using a *bit reversal* algorithm.

The *bit reversal* algorithm used to perform this reordering is shown in Table 9.6. The decimal index, n, is converted to its binary equivalent. The binary bits are then placed in reverse order, and converted back to a decimal number. Bit reversing is often performed in DSP hardware in the data address generator (.*), thereby simplifying the software, reducing overhead, and speeding up the computations.

Table 9.6 Bit reversal example for $N = 8$

Decimal number:	0	1	2	3	4	5	6	7
Binary equivalent:	000	001	010	011	100	101	110	111
Bit-reversed binary:	000	100	010	110	001	101	011	111
Decimal equivalent:	0	4	2	6	1	5	3	7

The computation of FFT using *decimation-in-frequency* (DIF) is shown in Figs. 9.25 and 9.26. This method requires that the bit reversal algorithm be applied to the output $X(k)$. Note that the butterfly for the DIF algorithm differs slightly from the decimation-in-time butterfly, as shown in Fig. 9.27.

The use of decimation-in-time versus decimation-in-frequency algorithms is largely a matter of preference, as either yields the same result. System constraints may make one of the two a more optimal solution. It should be noted that the algorithms required to compute the inverse FFT are nearly identical to those required to compute the FFT, assuming complex FFTs are used. In fact, a useful method for verifying a complex FFT algorithm consists of first taking the FFT of the $x(n)$ time samples and then taking the inverse FFT of the $X(k)$. At the end of this process, the original time samples, $Re(x(n))$, should be obtained and the imaginary part, $Im(x(n))$, should be zero (within the limits of the mathematical round off errors).

The FFTs discussed up to this point are radix-2 FFTs, i. e., the computations are based on two-point butterflies. This implies that the number of points in the FFT algorithms must be a power of 2. However, non-radix-2 FFT algorithms have been developed and are available in modern computational software packages like Matlab, Mathematica, Maple, etc., to be used in a variety of applications in ocean engineering as well as beyond [9.1, 2, 4–6].

9.4 Waveform Analysis

A waveform is one recording of a deterministic signal or that of an instance of a random signal (also referred to in the literature as a stochastic process). Actually, in practice we can make a recording of finite duration, which means that a waveform is the set of signal recordings over possibly non-contiguous time intervals. Typical examples of (lumped) waveforms in ocean engineering are significant wave height at a point or average, oceanic temperature or salinity, acoustic recordings of marine life, sonar or radar signals, etc. [9.3].

9.4.1 Definitions for Waveforms and Random Signals

The time mean (or average) of a waveform is defined as follows in the continuous time domain

$$\mu_x \triangleq \lim_{T\to\infty}\left\{\frac{1}{2T}\int_{-T}^{+T} x(t)\,\mathrm{d}t\right\} . \tag{9.74}$$

The time-wise cross-correlation of two waveforms is defined as follows

$$\begin{aligned} R_{y,x}(\tau) &\triangleq \lim_{T\to\infty}\left\{\frac{1}{2T}\int_{-T}^{+T} y(t)x(t+\tau)\,\mathrm{d}t\right\} ,\\ &= [y \otimes x_{-(t)}](\tau) . \end{aligned} \tag{9.75}$$

In the above, $\otimes$ stands for the convolution operation applied in the square brackets between signal y and signal $x_{(-t)}$, which is the mirrored version of signal x about origin of time ($t=0$), with time shift τ, i. e.,

$$x_{-(t)}(t) = x(-t) .$$

Also, note that cross-correlation as defined above can be used to introduce (time-wise) autocorrelation of a single waveform if we set $y \equiv x$.

The limiting process to infinity for time variable T introduced in the equations above is required in the case when the waveform has infinite duration so that the definitions make sense and do not give rise to indeterminate or infinite sums.

On the other hand, if the waveform (or at least one of them in the cross-correlation case) is finite in duration, then the time interval ($2T$) needs to be set equal to the full finite, and not infinite, duration of the signal. In this case, the definitions need to look like the following to avoid any issues

$$\mu_x \triangleq \frac{1}{2T}\int_{-\infty}^{+\infty} x(t)\,\mathrm{d}t , \tag{9.76}$$

$$R_{y,x}(\tau) \triangleq \frac{1}{2T}\int_{-\infty}^{+\infty} y(t)x(t+\tau)\,\mathrm{d}t . \tag{9.77}$$

The above equations are well defined in the case of deterministic signals and waveforms that occur as instances of random signals. Specifically, a random signal, denoted with a capital letter, e.g., $X(t)$, in contrast to lowercase latters used for deterministic signals, is associated with a probability density function (PDF), $f_X(x)$ quantifying the probability of the signal value at any time instant t to lie within an arbitrary real interval $[a, b]$ as follows

$$\Pr[a \leq X(t) \leq b] = \int_a^b f_X(x)\,\mathrm{d}x\,. \tag{9.78}$$

As well established by probability theory, the probability density function is, in turn, associated with a cumulative distribution function (CDF) for random signal $X(t)$ as follows

$$F_X(x) = \Pr[X(t) \leq x] = \int_{-\infty}^{x} f_X(x_\#)\,\mathrm{d}x_\#\,. \tag{9.79}$$

One can easily verify by combining (9.78) with (9.79) that

$$\Pr[a \leq X(t) \leq b] = F_X(b) - F_X(a) \tag{9.80}$$

holds. Finally and assuming certain smoothness conditions hold, the PDF can be calculated as the derivative of the CDF for the same random variable, i. e.,

$$f_X(x) = \frac{\mathrm{d}}{\mathrm{d}x} F_X(x)\,. \tag{9.81}$$

For a random signal one can, in effect, define a probabilistic mean value as follows

$$\langle X(t)\rangle = \mu_X(t) \triangleq \int_{-\infty}^{+\infty} x f_{X(t)}(x)\,\mathrm{d}x\,. \tag{9.82}$$

Along the same lines a probabilistic correlation (cross and/or autocorrelation) can be introduced

$$\begin{aligned}\langle X(t_1), Y(t_2)\rangle &= R_{X,Y}(t_1, t_2)\,,\\ &\triangleq \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} xy f_{X(t_1),Y(t_2)}(x,y)\,\mathrm{d}y\,\mathrm{d}x\,.\end{aligned} \tag{9.83}$$

To properly define it, though, the joint probability density function needs to be introduced; as can be seen below, it quantifies the probability of random signal X assuming a value in an infinitesimal vicinity of x at instant t_1 and, jointly, random signal Y assuming a value in an infinitesimal vicinity of y at instant t_2

$$\begin{aligned}&f_{X(t_1),Y(t_2)}(x,y)\,\mathrm{d}x\,\mathrm{d}y\\ &= \Pr[(x \leq X(t_1) \leq x + \mathrm{d}x) \text{ and}\\ &\qquad (y \leq Y(t_2) \leq y + \mathrm{d}y)]\,.\end{aligned} \tag{9.84}$$

In effect, the probability on the right-hand side in the above is that of the joint event to find the values of signals X and Y at certain time instants within an arbitrary infinitesimal rectangle on the XY plane. Applying Bayes' theorem on the joint probability of the right-hand side of (9.84) one can derive the following equations involving conditional probabilities

$$\begin{aligned}&\Pr[x \leq X(t_1) \leq x + \mathrm{d}x | y \leq Y(t_2) \leq y + \mathrm{d}y]\\ &\quad \cdot \Pr[y \leq Y(t_2) \leq y + \mathrm{d}y]\\ &= f_{X(t_1),Y(t_2)}(x,y)\,\mathrm{d}x\,\mathrm{d}y\\ &= \Pr[y \leq Y(t_2) \leq y + \mathrm{d}y | x \leq X(t_1) \leq x + \mathrm{d}x]\\ &\quad \cdot \Pr[x \leq X(t_1) \leq x + \mathrm{d}x]\,.\end{aligned} \tag{9.85}$$

In the case that random variables X and Y are independent (not to be confused with mutually exclusive variables or events), then using (9.85) for the conditional probabilities one can see that the following holds

$$f_{X(t_1),Y(t_2)}(x,y) = f_{X(t_1)}(x) \cdot f_{Y(t_2)}(y) \tag{9.86}$$

A very important category of random signals is that of wide sense stationary (WSS). A WSS signal is one that has: (a) constant, i. e., time invariant, probabilistic mean, and, (b) autocorrelation solely depending on the time shift (delay) $\tau = t_2 - t_1$ and not on the individual time instants. In effect, for a random signal to be WSS it must hold that

$$\langle X(t)\rangle = \langle X\rangle = \mu_X\,, \tag{9.87}$$

$$\langle X(t), X(t+\tau)\rangle = R_{X,X}(\tau)\,. \tag{9.88}$$

In part, WSS signals are important because both the probabilistic mean value and autocorrelation can be obtained as the time-wise mean and autocorrelation of anyone of its instances, $x(t)$, introduced for waveforms earlier. For this to be possible though, the random signal (stochastic process) X needs to be ergodic on top of WSS. Indeed, for an ergodic process statistical (probabilistic) properties (such as its mean, correlation, and variance) can be deduced from a single, sufficiently long instance (sample realization) of the process. Therefore, for an ergodic WSS process, the following hold

$$\langle X(t)\rangle = \mu_x\,, \tag{9.89}$$

$$\langle X(t), X(t+\tau)\rangle = R_{x,x}(\tau)\,. \tag{9.90}$$

It should be noted, however, that for the above equations to hold exactly the integrals over time in (9.74) and (9.75) need to span the entire real line from minus to plus infinity. If not, then (9.89) and (9.90) provide only an estimate of the mean and autocorrelation values of the random signal; the accuracy of the estimate (probabilistically) increases as time parameter T in the time-wise integrals of (9.74) and (9.75) grow larger.

A final note is made regarding strictly or strongly stationary processes versus WSS ones. A strictly stationary process is a stochastic process whose joint probability distribution does not change when shifted in time. This requirement is much stronger than just mean and autocorrelation and, therefore, much harder to meet or assume in practical situations; that is why in the remaining text we will only employ WSS as well as ergodicity.

9.4.2 Signal Power and Power Spectral Density

For a waveform $x(t)$, i. e., a deterministic signal or an instance of a random one, the following definitions are given:

1. The time mean is the direct current (DC) offset (component) of the waveform and $|\mu_x|^2$ is the DC power of it.
2. Consider

$$R_{x,x}(0) = \lim_{T\to\infty}\left\{\frac{1}{2T}\int_{-T}^{+T} x^2(t)\,\mathrm{d}t\right\} \geq 0\,. \qquad (9.91)$$

   This is the (total averaged) power of the waveform.
3. Then consider

$$\operatorname{cov}(x) = R_{x,x}(0) - |\mu_x|^2 \geq 0\,. \qquad (9.92)$$

   This is the covariance or alternating current (AC) power of the waveform. Also, $[\operatorname{cov}(x)]^{1/2}$ is the standard deviation or RMS (root-mean-square) value of the waveform.

The use of term *power* here stands for electric power, i. e., that the waveform is considered as voltage across or electric current through a reference resistance equal to 1 ohm.

Finally, the following notation for cross-correlation (for two waveforms) or autocorrelation (for a single waveform)

$$\langle y, x\rangle = R_{y,x}(0) \triangleq \lim_{T\to\infty}\left\{\frac{1}{2T}\int_{-T}^{+T} y(t)x(t)\,\mathrm{d}t\right\}\,. \qquad (9.93)$$

In the case of two distinct waveforms $\langle y, x\rangle$ is also identified as their dot product. In the case of a single waveform $\langle x, x\rangle$ is its power.

In the frequency domain and using the two-sided Fourier transform as introduced in (9.8), the (cross) power spectral density (PSD) can be defined for a couple of waveforms as follows

$$S_{y,x}(\omega) \triangleq \mathcal{F}\{R_{y,x}(\tau)\} = Y(\omega)X^*(\omega)\,. \qquad (9.94)$$

Note that for a real waveform it holds that

$$X(\omega) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t)\exp(-i\omega t)\,\mathrm{d}t$$

$$\Updownarrow$$

$$\begin{aligned} X^*(\omega) &= \left(\int_{-\infty}^{+\infty} x(t)\exp(-i\omega t)\,\mathrm{d}t\right)^* \\ &= \int_{-\infty}^{+\infty} x^*(t)\exp(+i\omega t)\,\mathrm{d}t = X(-\omega)\,. \end{aligned}$$

Furthermore

$$\begin{aligned} \mathcal{F}\{x(-t)\} &= \int_{-\infty}^{+\infty} x(-t)\exp(-i\omega(-t))\,\mathrm{d}(-t) \\ &= \int_{-\infty}^{+\infty} x(t)\exp(i\omega t)\,\mathrm{d}t = X(-\omega)\,. \end{aligned}$$

Therefore, for a real waveform it holds that

$$X^*(\omega) = \mathcal{F}\{x(-t)\} = X(-\omega)\,. \qquad (9.95)$$

Then, using (9.94) one can directly derive the following

$$\begin{aligned} &S_{x,y}(\omega) = S_{y,x}(-\omega) = S^*_{y,x}(\omega) \\ &\quad\Leftrightarrow R_{x,y}(\tau) = R_{y,x}(-\tau)\,. \end{aligned} \qquad (9.96)$$

Moreover, in the case of autocorrelation PSD obtains the following form

$$\begin{aligned} &S_{x,x}(\omega) \triangleq \mathcal{F}\{R_{x,x}(\tau)\} = |X(\omega)|^2 \geq 0 \quad \text{and} \\ &S_{x,x}(\omega) = S_{x,x}(-\omega)\,. \end{aligned} \qquad (9.97)$$

In effect, the PSD of a single signal is an even, nonnegative complex function of frequency. Also, for the autocorrelation it holds that it is an even function of time τ, i. e., $R_{x,x}(\tau) = R_{x,x}(-\tau)$.

A final note has to do with the extension of the PSD concept to random signals (stochastic processes in mathematical terminology). Assuming ergodicity and WSS, the Wiener–Khinchin theorem can be employed to introduce a PSD for a random signal X or at least a power spectral distribution function $F_X(f)$ as follows

$$\begin{aligned}\langle X(t), X(t+\tau)\rangle &= R_{X,X}(\tau)\\ &= \int\limits_{-\infty}^{+\infty} e^{i\omega\tau}\,\mathrm{d}F_X(f), \omega = 2\pi f\,.\end{aligned} \tag{9.98}$$

For the above to hold, the autocorrelation of X needs to exist and be finite for any value of time shift variable τ, but its Fourier transform may not be well defined. Indeed, the Fourier transform of a random signal may not exist in general, because stationary stochastic processes may not generally be square or absolutely integrable. Nor does their autocorrelation need to be absolutely integrable, so it need not have a Fourier transform, either. However, in most practical applications the autocorrelation is integrable and even satisfies the conditions to obtain a Fourier transform. In this case, the PSD can be introduced through the Fourier transform as follows

$$\begin{aligned}S_{X,X}(\omega) &= \int\limits_{-\infty}^{+\infty} R_{X,X}(\tau)e^{-i\omega\tau}\,\mathrm{d}\tau \Leftrightarrow R_{X,X}(\tau)\\ &= \int\limits_{-\infty}^{+\infty} S_{X,X}(f)e^{i\omega\tau}\,\mathrm{d}f, \omega = 2\pi f\,.\end{aligned} \tag{9.99}$$

Finally, in this case the PSD is the averaged derivative of the power spectral distribution function introduced previously. This is why the latter is also referred to as the integrated spectrum of the stochastic process.

9.4.3 Waveform Propagation Through a Linear, Time-Invariant System

Consider a linear, time-invariant (LTI) system with a single input and a single output (SISO), as shown in Fig. 9.28, driven by input waveform $x(t)$ and generating as response output waveform $y(t)$.

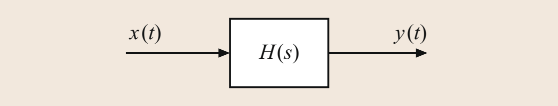


Fig. 9.28 LTI-SISO system with waveforms at its input and its output

The system is assumed to be in continuous time, as well as the waveforms at its input and its output. Then, the output can be determined as the convolution ($\otimes$) of the LTI system's scalar impulse response with the input waveform

$$\begin{aligned}y(t) = h(t)\otimes x(t) &= \int\limits_{-\infty}^{+\infty} h(t_\#)x(t-t_\#)\,\mathrm{d}t_\#\\ &\Leftrightarrow Y(\omega) = H(\omega)\cdot X(\omega)\,.\end{aligned} \tag{9.100}$$

In the above, $h(t)$ stands for the system's impulse response that can be determined on the basis of the system's transfer function $H(s)$ in the complex frequency (Laplace transform) domain

$$h(t) = \mathcal{L}^{-1}\{H(s)\}\,. \tag{9.101}$$

It is noted here that for a causal system, the convolutional integral must be finitely bounded as follows

$$\begin{aligned}y(t) = h(t)\otimes x(t) &= \int\limits_{0}^{t} h(t_\#)x(t-t_\#)\,\mathrm{d}t_\#\\ \text{and}\quad & h(t) = 0, \forall t < 0\,.\end{aligned} \tag{9.102}$$

For the means of the input and output waveforms the following can be derived

$$\begin{aligned}\mu_y &\triangleq \lim_{T\to\infty}\left\{\frac{1}{2T}\int\limits_{-T}^{+T} y(t)\,\mathrm{d}t\right\}\\ &= \mu_x\cdot\int\limits_{-\infty}^{+\infty} h(t)\,\mathrm{d}t = H(0)\cdot\mu_x\,.\end{aligned} \tag{9.103}$$

For the various correlations, as well as the PSD of the input and output, the following hold

$$\begin{aligned}R_{y,y}(t) &= R_{h,h}\otimes R_{x,x}\\ &\Leftrightarrow S_{y,y}(\omega) = |H(\omega)|^2\cdot S_{x,x}(\omega)\,,\end{aligned} \tag{9.104}$$

$$\begin{aligned}R_{y,x}(t) &= h\otimes R_{x,x}\\ &\Leftrightarrow S_{y,x}(\omega) = H(\omega)\cdot S_{x,x}(\omega)\,,\end{aligned} \tag{9.105}$$

$$\begin{aligned} R_{x,y}(t) &= h_{-(t)} \otimes R_{x,x} \\ &\Leftrightarrow S_{x,y}(\omega) = H^{*}(\omega) \cdot S_{x,x}(\omega)\,. \end{aligned} \tag{9.106}$$

All of the above can be derived on the basis of the definitions and properties given previously, as well as the commutativity and associativity of scalar convolution. For example, in the case of (9.104)

$$\begin{aligned} R_{y,y}(t) &= y \otimes y_{-(\mathrm{t})} \\ &= [h \otimes x] \otimes [h_{-(t)} \otimes x_{-(t)}] \\ &= R_{h,h} \otimes R_{x,x}\,. \end{aligned} \tag{9.107}$$

It is noted here that:

1.

$$S_{h,h}(\omega) = H(\omega)H^{*}(\omega) = |H(\omega)|^{2}\,,$$

2.

$$\begin{aligned} R_{h,h}(\tau) &= \int_{-\infty}^{+\infty} S_{h,h}(f) e^{i\omega\tau}\,\mathrm{d}f \\ &= \int_{-\infty}^{+\infty} |H(\omega)|^{2} e^{i\omega\tau}\,\mathrm{d}f, \omega = 2\pi f\,, \end{aligned}$$

and

3.

$$\begin{aligned} \begin{pmatrix} x_{-(\mathrm{t})} \\ y_{-(\mathrm{t})} \\ h_{-(\mathrm{t})} \end{pmatrix}(t) &= \begin{pmatrix} x \\ y \\ h \end{pmatrix}(-t), \forall t \in \mathbb{R} \Leftrightarrow \mathcal{F}\begin{Bmatrix} x_{-(\mathrm{t})} \\ y_{-(\mathrm{t})}(t) \\ h_{-(\mathrm{t})} \end{Bmatrix} \\ &= \begin{pmatrix} X^{*} \\ Y^{*} \\ H^{*} \end{pmatrix}(\omega \text{ or } f)\,. \end{aligned}$$

9.5 Optimal Signal Estimation

9.5.1 System Identification

One of the most fundamental problems in science and engineering is the determination of a system's dynamics, preferably in the form of a mathematical model, when presented with the system response(s) to given deterministic or random driving inputs. This is the fundamental system identification problem [9.7, 8].

In the case of an LTI system, especially in the case of discrete-time or sampled-data systems, system identification is essentially equivalent to a problem of minimum square error approximation. Consider the problem statement as shown in Fig. 9.29.

A fundamental prerequisite for the system identification problem to have a solution in its basic form is that the additive measurement noise superimposed to the system output is of zero mean and uncorrelated to the driving forcing signal applied as input to the system. Therefore, the following two conditions must hold

$$\mu_{v} \triangleq \lim_{T\to\infty}\left\{\frac{1}{2T}\int_{-T}^{+T} v(t)\,\mathrm{d}t\right\} = 0\,, \tag{9.108}$$

$$\begin{aligned} R_{vx}(\tau) &= R_{xv}(\tau) = 0, \forall\tau \\ &\Leftrightarrow S_{vx}(\omega) = S_{xv}(\omega) = 0, \forall\omega\,. \end{aligned} \tag{9.109}$$

At this point it is important to emphasize the importance of the definition of the appropriate time window T in (9.75). Indeed, it is possible that for some (commonly small) value of T condition (9.109) may not be satisfied with sufficient accuracy. However, assuming that condition (9.109) is satisfied for an infinite time window as well as appropriate stationarity and ergodicity assumptions, it is possible to determine a finite value for T such that condition (9.109) is met at arbitrarily small (*epsilon*) accuracy.

Given the property in (9.109) and by use of the linearity property of both cross-correlation and PSD, as well as (9.105), one can derive the following

$$S_{yx}(\omega) = S_{ux}(\omega) + S_{vx}(\omega) = H(\omega) \cdot S_{xx}(\omega)\,. \tag{9.110}$$

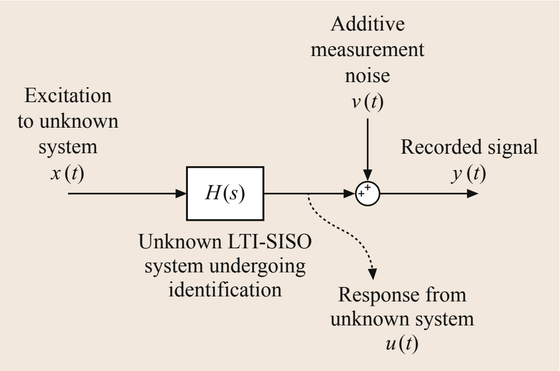


Fig. 9.29 LTI-SISO system identification problem

The above holds since $y(t) = u(t) + v(t)$ and $u(t) = h(t) \otimes x(t) \Leftrightarrow U(\omega) = H(\omega)X(\omega)$. In effect, the unknown transfer functions of the intermitted LTI system

$$H(\omega) = \frac{S_{yx}(\omega)}{S_{xx}(\omega)} . \tag{9.111}$$

In the time domain the equivalent equation to (9.111) is

$$R_{yx}(t) = h(t) \otimes R_{xx}(t) = \int_{-\infty}^{+\infty} h(t_\#) R_{xx}(t - t_\#)\, \mathrm{d}t_\# . \tag{9.112}$$

In the above equation, the unknown signal (function) is the impulse, $h(t)$. The integral equation is known as the Wiener–Hopf equation with infinite time horizon [9.1, 2, 4–6].

What is most important in both (9.111) and (9.112) is that system identification using them can be implemented with either deterministic or random signals. In effect, signals, $x(t)$, $v(t)$ and $y(t)$ may be known only probabilistically (statistically); this means that the individual instances (realizations) of the signals do not need to be known as long as their statistics are, e.g., means, correlations, or covariance coefficients.

Typically in practice, statistical characterization of signals can be effectively, yet approximately, performed by taking advantage of ergodicity and stationarity, if they apply. Assuming that these hold, then a sufficiently long, yet finite, recording of a single realization of a random signal allows for its statistical characterization in terms of its autocorrelation and cross-correlation with other signals. Such a recording, e.g., for $v(t)$ or $x(t)$ in the case of our system identification framework, can be performed once and be used along with measurements for $y(t)$ to identify the system; all due to ergodicity and stationarity.

9.5.2 Discrete-Time Wiener–Hopf Equation over a Finite-Duration Window

Integral (9.112), despite its immense theoretical value, does not really indicate how it can be applied in practice for system identification. In this passage, we will introduce a methodology allowing identification of an LTI-SISO system when presented only with sampled-data recordings of $x(t)$ and $y(t)$, with a sampling rate meeting the Nyquist requirement for both. Furthermore, it will be assumed that the observation window (time horizon) of the recordings of the forcing and the noisy response is finite in duration instead of infinite.

The outcome of the processing will be the determination of a given finite number of impulse response samples for the unknown system undergoing identification. In effect, it is possible to define a discrete-system FIR (finite impulse response) system in either the time domain, by its very impulse response, or as a transfer function in the Z-transform domain without any poles. In effect, the FIR system will approximate the behavior of the unknown LTI system undergoing identification in the sense of (truncated) impulse response matching.

We start from the integral equation for $y(t)$ given previously

$$\begin{aligned} y(t) &= h(t) \otimes x(t) + v(t) \\ &= \int_{-\infty}^{+\infty} h(t_\#) x(t - t_\#)\, \mathrm{d}t_\# + v(t) . \end{aligned} \tag{9.113}$$

However, to satisfy causality for the unknown system the following needs to be introduced

$$\begin{aligned} y(t) &= \int_{0}^{t} h(t_\#) x(t - t_\#)\, \mathrm{d}t_\# \\ +v(t) &= \int_{0}^{t} h(t - t_\#) x(t_\#)\, \mathrm{d}t_\# + v(t) . \end{aligned} \tag{9.114}$$

Converting the above to discrete time by sampling interval T_s, compliant to the Nyquist rate criterion, one can derive the following

$$y(nT_\mathrm{s}) = T_\mathrm{s} \sum_{k=0}^{n} h(kT_\mathrm{s}) x((n-k)T_\mathrm{s}) + v(nT_\mathrm{s}) . \tag{9.115}$$

Without loss of generality we can assume that $T_\mathrm{s} = 1$, since any other sampling interval value can be absorbed multiplicatively in the impulse response sample values.

$$y(n) = \sum_{k=0}^{n} h(k) x(n-k) + v(n) . \tag{9.116}$$

An important prerequisite for our identification method to be successful is that the unknown LTI system is stable. If this holds, then for an arbitrarily small ε one can find a positive integer M, such that

$$\sum_{n=M+1}^{+\infty} h^2(n) < \varepsilon . \tag{9.117}$$

In effect, with epsilon accuracy it is possible to truncate the impulse response of the unknown system. We note here that impulse response truncation is an approximation of the system undergoing identification by the following FIR one with memory M

$$\hat{H}_{\text{FIR}}(z^{-1})=\sum_{n=0}^{M}h(n)z^{-n}\,. \tag{9.118}$$

Using the approximation above, (9.116) yields

$$y(n)=\sum_{k=0}^{M}h(k)x(n-k)+v(n)\,. \tag{9.119}$$

Furthermore, assuming that the recording for waveform $y(t)$ spans a finite interval $[0,T]$, the identification problem can be reduced to determining the $(M+1)$ samples of the impulse response, $h(n)$, $0\leq n\leq M$, of the unknown system so that the following $(L+1)$ equations are satisfied

$$y(n)=\sum_{k=0}^{M}h(k)x(n-k)+v(n),0\leq n\leq L$$
$$\text{and}\quad L=\frac{T}{T_{\text{s}}}\,. \tag{9.120}$$

It is noted here that containing the recording of signal $y(t)$ within interval $[0,T]$ allows limiting the need to record signal $x(t)$ also in the finite interval $[-MT_{\text{s}},+LT_{\text{s}}]$.

In effect, the $((L+1)\times(M+1))$ set of algebraic (9.120) can be put in the following matrix form

$$\boldsymbol{y}=\ddot{X}\cdot\boldsymbol{h}+\boldsymbol{v}\,. \tag{9.121}$$

In the above, vectors y and v are defined as follows

$$\begin{aligned}\boldsymbol{y}&=[y(0)\quad y(1)\quad\cdots\quad y(L)]^{\text{T}}\\ \boldsymbol{v}&=[v(0)\quad v(1)\quad\cdots\quad v(L)]^{\text{T}}\,.\end{aligned} \tag{9.122}$$

Vector h is defined as follows

$$\boldsymbol{h}=[h(0)\quad h(1)\quad\cdots\quad h(M)]^{\text{T}}\,. \tag{9.123}$$

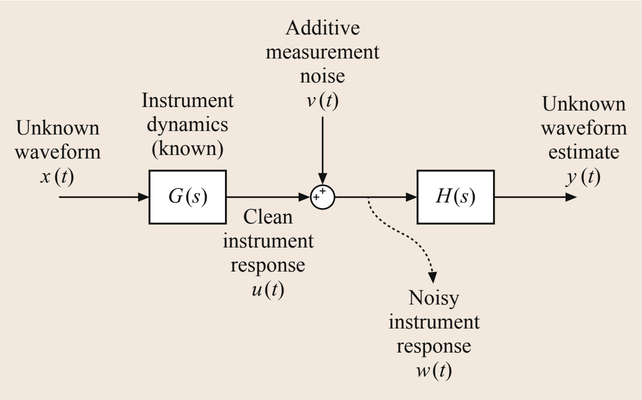

Additive measurement noise $v(t)$
Instrument dynamics (known)
Unknown waveform $x(t)$
$G(s)$
+
Clean instrument response $u(t)$
$H(s)$
Unknown waveform estimate $y(t)$
Noisy instrument response $w(t)$

Fig. 9.30 Signal estimation framework

Matrix $\mathbf{X}$, known as observation matrix, is defined by the following

$$\mathbf{X}=\begin{bmatrix}x(0) & x(-1) & \cdots & x(-M)\\ x(1) & x(0) & \cdots & x(-M+1)\\ \vdots & \vdots & \ddots & \vdots\\ x(L) & x(L-1) & \cdots & x(-M+L)\end{bmatrix}\,. \tag{9.124}$$

The equation set in (9.121) is not square; therefore, an algebraically exact solution is not feasible. In effect, only a solution minimizing the following mean square error (MSE) vector spanning the observation window

$$\|e\|_2=\|y-X\cdot h\|_2\,. \tag{9.125}$$

If the above MSE vector is minimized, then the effect of the noise vector v, i. e., signal $v(t)$, is removed from the measurement vector y, i. e., signal $y(t)$. The Euclidean norm of vector e is defined as follows

$$\|e\|_2=e^{\text{T}}\cdot e=\sum_{i=0}^{L}e_i^2\,. \tag{9.126}$$

The solution achieving minimization of $\|e\|_2$ can be proven to be that of the following square $((M+1)\times(M+1))$ set of algebraic *canonical equations*

$$(\mathbf{X}^T\cdot\mathbf{X})\cdot h=\mathbf{X}^T\cdot y\Rightarrow h=(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^Ty\,. \tag{9.127}$$

It is interesting to observe the equivalence between the time-domain (9.127) and (9.112); the equivalence can be straightforwardly derived if one observes that (9.112) needs to hold for any time instant t. This means that if it were converted to discrete time a (countable infinite in number) set of algebraic equations would be derived like in (9.121). Furthermore, the solution of canonical (9.127) is fully equivalent to (9.111), since there exists an one-to-one correspondence between: (a) square matrix $(\mathbf{X}^{\text{T}}\mathbf{X})$ and autocorrelation $R_{xx}(t)$, and thereof PSD $S_{xx}(\omega)$; (b) vector $(\mathbf{X}^{\text{T}}y)$ and cross-correlation $R_{yx}(t)$, and thereof PSD $S_{yx}(\omega)$.

On a technical note, it is pointed out that for (9.127) to hold, matrix $(\mathbf{X}^{\text{T}}\mathbf{X})$ has to be invertible. In the case of zero initial conditions for x, i. e., $x(n)=0$, $-M\leq n<0$, it must hold that $x(0)\neq0$ so that $(\mathbf{X}^{\text{T}}\mathbf{X})$ is, indeed, invertible.

9.5.3 Signal Estimation and the Wiener Filter

We will now look into a closely related problem: that of the estimation of a signal or waveform which is outlined in the block diagram of Fig. 9.30.

As can be seen a known, in this case, the LTI-SISO system is driven by an unknown (to be estimated) $x(t)$. or the known LTI system, the transfer function $G(s)$, in the complex frequency (Laplace) domain, or $G(\omega)$, in the Fourier (frequency) domain, or equivalently its impulse response $g(t)$ in the time domain is assumed known. Such an LTI system may model a measurement instrument, or sensor, or transducer, or telecommunications receiver.

Then, a linear filter with transfer function $H(s)$ is sought that will receive as input the noisy instrument response $w(t)$ and output waveform $y(t)$, which is expected to be the optimum, in the mean square sense, estimate of the unknown waveform $x(t)$.

The objective is to determine transfer function $H(s)$, or equivalently $H(\omega)$, of the estimation filter that minimizes the following *objective cost function*

$$\langle e, e\rangle \triangleq R_{ee}(0) = \int_{-\infty}^{+\infty} e^2(t)\,\mathrm{d}t\,. \tag{9.128}$$

In the above, the instantaneous error signal is defined as follows

$$e(t) = x(t) - y(t)\,. \tag{9.129}$$

For waveform $y(t)$ it holds that

$$\begin{aligned} y(t) = h(t) \otimes w(t) &= \int_{-\infty}^{+\infty} h(\xi) w(t-\xi)\,\mathrm{d}\xi \\ &\approx T_\mathrm{s} \sum_{k=-\infty}^{+\infty} h(kT_\mathrm{s}) w(t-kT_\mathrm{s})\,. \end{aligned} \tag{9.130}$$

In the above, the convolutional integral has been approximated by the convolutional sum, assuming that sampling interval, T_s, satisfies the Nyquist criterion. Consequently, (9.128) becomes

$$\langle e, e\rangle = \int_{-\infty}^{+\infty} \left\{ x(t) - T_\mathrm{s} \sum_{k=-\infty}^{+\infty} h(kT_\mathrm{s}) w(t-kT_\mathrm{s}) \right\}^2 \mathrm{d}t\,. \tag{9.131}$$

In effect, the impulse response of the least-squares estimator has to fulfill the following condition, so that cost $\langle e, e\rangle$ becomes minimum

$$\frac{\partial}{\partial[h(nT_\mathrm{s})]}\langle e, e\rangle = 0, \forall n \in Z\,. \tag{9.132}$$

However,

$$\begin{aligned} &\frac{\partial}{\partial[h(nT_\mathrm{s})]}\langle e, e\rangle \\ &= -2 \int_{-\infty}^{+\infty} \left\{ x(t) - T_\mathrm{s} \sum_{k=-\infty}^{+\infty} h(kT_\mathrm{s}) w(t-kT_\mathrm{s}) \right\} \\ &\quad \times w(t-nT_\mathrm{s})\,\mathrm{d}t\,. \end{aligned} \tag{9.133}$$

Therefore, condition (9.132) finally yields

$$\int_{-\infty}^{+\infty} e(t) w(t-nT_\mathrm{s})\,\mathrm{d}t = 0, \forall n \in \mathbb{Z}\,. \tag{9.134}$$

Setting $\tau = nT_\mathrm{s}$ in the above and gradually diminishing T_s, we end up back in the continuous time domain, and (9.134) yields the following *orthogonality condition*

$$\begin{aligned} &\int_{-\infty}^{+\infty} e(t) w(t-\tau)\,\mathrm{d}t = 0, \forall \tau \in \mathbb{R} \\ &\qquad \Leftrightarrow R_{we}(\tau) = 0, \forall \tau \in \mathbb{R}\,. \end{aligned} \tag{9.135}$$

Using the above orthogonality condition

$$R_{we}(\tau) = 0 \Leftrightarrow S_{we}(\omega) = 0 \Leftrightarrow S_{xw}(\omega) = S_{yw}(\omega)\,. \tag{9.136}$$

Furthermore, since $S_{yw}(\omega) = H(\omega) S_{ww}(\omega)$, the following equation can be derived for the transfer function of the estimation filter

$$H(\omega) = \frac{S_{xw}(\omega)}{S_{ww}(\omega)}\,. \tag{9.137}$$

The above expression for the estimator is commonly known as the Wiener filter and of is of widespread use in many science and engineering application fields.

The signal power of the minimum square error, achieved by the Wiener filter in (9.137), can be determined as follows

$$\begin{aligned} S_{ee}(\omega) &= S_{xe}(\omega) - \underbrace{S_{ye}(\omega)}_{0} = S_{xx}(\omega) - S_{xy}(\omega) \\ &= S_{xx}(\omega) - \left(S_{yy}(\omega) + \underbrace{S_{ey}(\omega)}_{0} \right) \\ &\Downarrow \\ S_{ee}(\omega) &= S_{xx}(\omega) - S_{yy}(\omega)\,. \end{aligned} \tag{9.138}$$

For the derivation of the above it is reminded that

$$e = x - y \, .$$

Also, using the orthogonality condition (9.135), the following important result can be derived

$$S_{ye}(\omega) = H(\omega) S_{we}(\omega) \overset{S_{we}=0}{\Rightarrow} S_{ye}(\omega) = 0 = S_{ey}(\omega) \, .$$

It is noted here that

$$R_{ee}(\tau) = \mathcal{F}^{-1}\{S_{ee}(\omega)\} = \int_{-\infty}^{+\infty} S_{ee}(\omega) e^{\mathrm{i}\omega\tau} \, \mathrm{d}\omega \Rightarrow \langle e, e \rangle$$
$$= R_{ee}(0) = \int_{-\infty}^{+\infty} S_{ee}(\omega) \, \mathrm{d}\omega \, .$$

An alternative expression for the minimum error PSD can be derived on the basis of the following

$$S_{yy}(\omega) = |H(\omega)|^2 S_{ww}(\omega) = \frac{|S_{xw}(\omega)|^2}{S_{ww}^2(\omega)} S_{ww}(\omega)$$
$$= \frac{|S_{xw}(\omega)|^2}{S_{ww}(\omega)} \, .$$

Then

$$S_{ee}(\omega) = S_{xx}(\omega) - S_{yy}(\omega)$$
$$= \frac{S_{xx}(\omega) S_{ww}(\omega) - |S_{xw}(\omega)|^2}{S_{ww}(\omega)} \, . \tag{9.139}$$

Further treatment of (9.137) is possible since the additive measurement noise is uncorrelated to the unknown signal $x(t)$ at the instrument input. In this case,

$$S_{xw}(\omega) = S_{xu}(\omega) + S_{xv}(\omega)$$
$$= G^*(\omega) S_{xx}(\omega) \quad \text{because} \quad S_{xv}(\omega) = 0 \, . \tag{9.140}$$

Also

$$S_{ww}(\omega) = S_{uu}(\omega) + S_{vv}(\omega)$$
$$= |G(\omega)|^2 S_{xx}(\omega) + S_{vv}(\omega)$$
$$\text{since } u = v + w \, . \tag{9.141}$$

Therefore, the Wiener filter in (9.137) becomes as follows

$$H(\omega) = \frac{G^*(\omega) S_{xx}(\omega)}{|G(\omega)|^2 S_{xx}(\omega) + S_{vv}(\omega)} \, . \tag{9.142}$$

The above allows us to obtain an estimate of waveform $x(t)$ appearing at the instrument input port. As can be seen, to achieve this the signal's (waveform) statistics need to be known in advance, i. e., the signal PSD $S_{xx}(\omega)$ or autocorrelation $R_{xx}(\tau)$, as well as the dynamics of the measurement instrument, i. e., its transfer function $G(\omega)$ or its impulse response $g(t)$; last, but not least, prior information of the additive noise's statistics, ie, its PSD $S_{vv}(\omega)$ or autocorrelation $R_{vv}(\tau)$.

Two important specific subcases of (9.137) are given below:

1. *Zero-forcing equalization* [9.1, 5]:
   If $S_{vv}(\omega) \ll S_{xx}(\omega)$

   $$H(\omega) = \frac{1}{G(\omega)} \, . \tag{9.143}$$

   Evidently, in this subcase the statistics of either the unknown input signal or the additive noise need to be known in advance. Furthermore, one can easily verify that, if the inverse of the instrument's transfer function $G(\omega)$ does not introduce unstable poles, estimation error e vanishes.
2. *Matched filter* [9.1, 5]:
   If $|G(\omega)|^2 S_{xx}(\omega) \ll S_{vv}(\omega)$

   $$H(\omega) = G^*(\omega) \frac{S_{xx}(\omega)}{S_{vv}(\omega)} \, . \tag{9.144}$$

   In this subcase the major impairment toward obtaining an estimate of $x(t)$ is the additive noise rather than the instrument distortion, as was the case previously. Moreover note that if $S_{xx}(\omega) < S_{vv}(\omega)$, the instrument's output $w(t)$ will be noise-like, and yet by use of the Wiener filter it is possible to extract the unknown *information* signal $x(t)$.

Matched filter estimator design is widely used in communications engineering. At least as a first approximation, transfer function $G(\omega)$ models the telecommunication channel that introduces attenuation monotonically increasing with distance from the transmitter source. In a typical system, both the information signal $x(t)$, as well as disturbance noise $v(t)$ can be considered as *white noise*, i. e., signals with the following property

$$S_{nn}(\omega) = N_0, \forall \omega \Leftrightarrow R_{nn}(\tau) = N_0 \delta(\tau), \forall \tau \, . \tag{9.145}$$

As can be seen, the case of a white noise signal demonstrates practically no predictability, since the value of the signal at any time instant is entirely uncorrelated with its value at any other time instant in the past or in the future. Therefore, if both the information signal and the additive noise are modeled as white noise, (9.144) becomes

$$H(\omega) = \frac{X_0}{V_0} G^*(\omega) \, . \tag{9.146}$$

The ratio (X_0/V_0) with the PSD of the information signal $x(t)$ in the numerator and that of the additive noise in the denominator is known as signal-to-noise ratio (SNR) and is a very important parameter to characterize a system's sensitivity to external noise, as well as exogenous disturbance and crosstalk interference [9.1, 5].

9.6 Concluding Remarks

In this chapter, an account of digital signal processing concepts, methods and techniques employed in ocean engineering is given. After looking into the fundamental processes of continuous signal sampling and reconstruction, the Z-transform is introduced as the tool of convenience to analyze difference equations just like the Laplace transform is suited to analyze differential equations. Digital filters are presented then as well as the Fast Fourier Transform algorithm. Fundamentals of waveform analysis and stochastic processes are presented last as employed for system identification and signal estimation.

References

9.1 J.G. Proakis: *Digital Communications*, 4th edn. (McGraw-Hill, New York 2000)

9.2 J.G. Proakis, D. Manolakis: *Digital Signal Processing: Principles, Algorithms and Applications*, 3rd edn. (Prentice Hall, Upper Saddle River 1995)

9.3 W.A. Kuperman, J.F. Lynch: Shallow-water acoustics, Phys. Today **57**(10), 55 (2004)

9.4 L. Ljung: *System Identification: Theory for the User*, 2nd edn. (Prentice Hall, Upper Saddle River 1999)

9.5 H.V. Poor, G.W. Wornell: *Wireless Communications: Signal Processing Perspectives* (Prentice Hall, Upper Saddle River 1998)

9.6 P.S.R. Diniz: *Adaptive Filtering: Algorithms and Practical Implementation*, 2nd edn. (Springer, Berlin, Heidelberg 2002)

9.7 T. Kohonen: *Self Organization and Associative Memory*, 3rd edn. (Springer Verlag, Berlin, Heidelberg 1989)

9.8 R.E. Schapire: *The Design and Analysis of Efficient Learning Algorithms* (MIT Press, Cambridge 1992)