The phase $\exp^{(j2\pi f t_0)}$ was added in order to center the impulse response in the time domain.

The real and imaginary components of the lowpass-equivalent impulse response $h(t)$ can be shown to be

$$h_1(t) = \frac{a+b}{2}\frac{\sin[\pi B(t-t_0)]}{\pi(t-t_0)}$$

$$h_2(t) = j\frac{a-b}{2}\frac{1-\cos[\pi B(t-t_0)]}{\pi(t-t_0)}$$

To demonstrate the simulation techniques described in previous sections, we present below the results of a simulation of the above filter for $a = 0.5$, $b = 1.0$, and the bandwidth $B = 1.0$. A delay $t_0 = T/2$ was added to center the impulse response.

To examine the frequency and impulse responses with high resolution, we used an FFT with 1024 samples and a sampling period of $T_s = 0.05$ s. This corresponds to an impulse response with duration $T = 50$s, a frequency span $F = 20$ Hz, and a sampling rate $f_s = 20$ samples/s. For most practical purposes such an idealized frequency response is not needed and often not desired because it produces long ringing tails in the time domain (Gibbs phenomenon distortion). We therefore truncated the impulse response to about five sidelobes on each side of $h_1(t)$ or a total of $T \approx 10$ s. Since the bandwidth of the filter is $B = 1.0$ Hz, a sampling rate $f_s = 5.0$ samples/s will be adequate. We chose, therefore, $T_s = 1/f_s = 0.2$ s and $T = NT_s = 12.9$ s, since it is desirable for the number of samples $N = 64$ to be an integer power of two, $N = 2^n$.

The truncated real and imaginary impulse responses are shown in Figure 4.18a. To be useful in a linear convolution (see Section 3.8.6), we augmented the impulse responses with an additional 64 zeros and performed the FFT with $N = 128$ samples. Figure 4.18b shows the resultant frequency response. As expected, the Gibbs phenomenon distortion is severe. It is interesting to note that the Gibbs phenomenon is not apparent in the 64-sample frequency response, which is due to the fact that the samples occur at the zero crossings of the frequency response (see Section 3.8.4.7 and Problem P4.8).

Figure 4.19a shows the impulse responses smoothed by a Hamming window

$$w(n) = 0.54 + 0.46\cos\left(\frac{2\pi n}{N}\right), \qquad -\frac{N-1}{2} \le n \le \frac{N-1}{2}$$

Since windowing is performed before augmenting with zeros, $N = 64$ in the equation above.

As seen from the frequency response of the windowed functions in Figure 4.19b, the Gibbs phenomenon distortion is reduced by about 20 dB. The frequency response, however, has slightly gentler rolloff.

The model of an FIR filter developed in this example can be used either for frequency- or for time-domain filtering. In the frequency domain it is used in a linear convolution type FFT filtering, using the appropriate technique. ■

4.1.2.4. Comparison of Time-Domain (Impulse Response) and Frequency-Domain (FFT) Implementations for FIR Filtering

The convolution sum is simple and straightforward to implement for FIR filters. Hence, time-domain convolution is the preferred type of filtering for FIR filters if the number of samples in the impulse response is not prohibitively large. As a rule of thumb, FIR filters requiring $N = 128$ samples or less may be implemented using the time-domain convolution method, without loss of efficiency.

Filters with sharp cutoffs generally have a "longer" impulse response, i.e., a larger interval over which the response is significant. Hence, to create an FIR filter there will

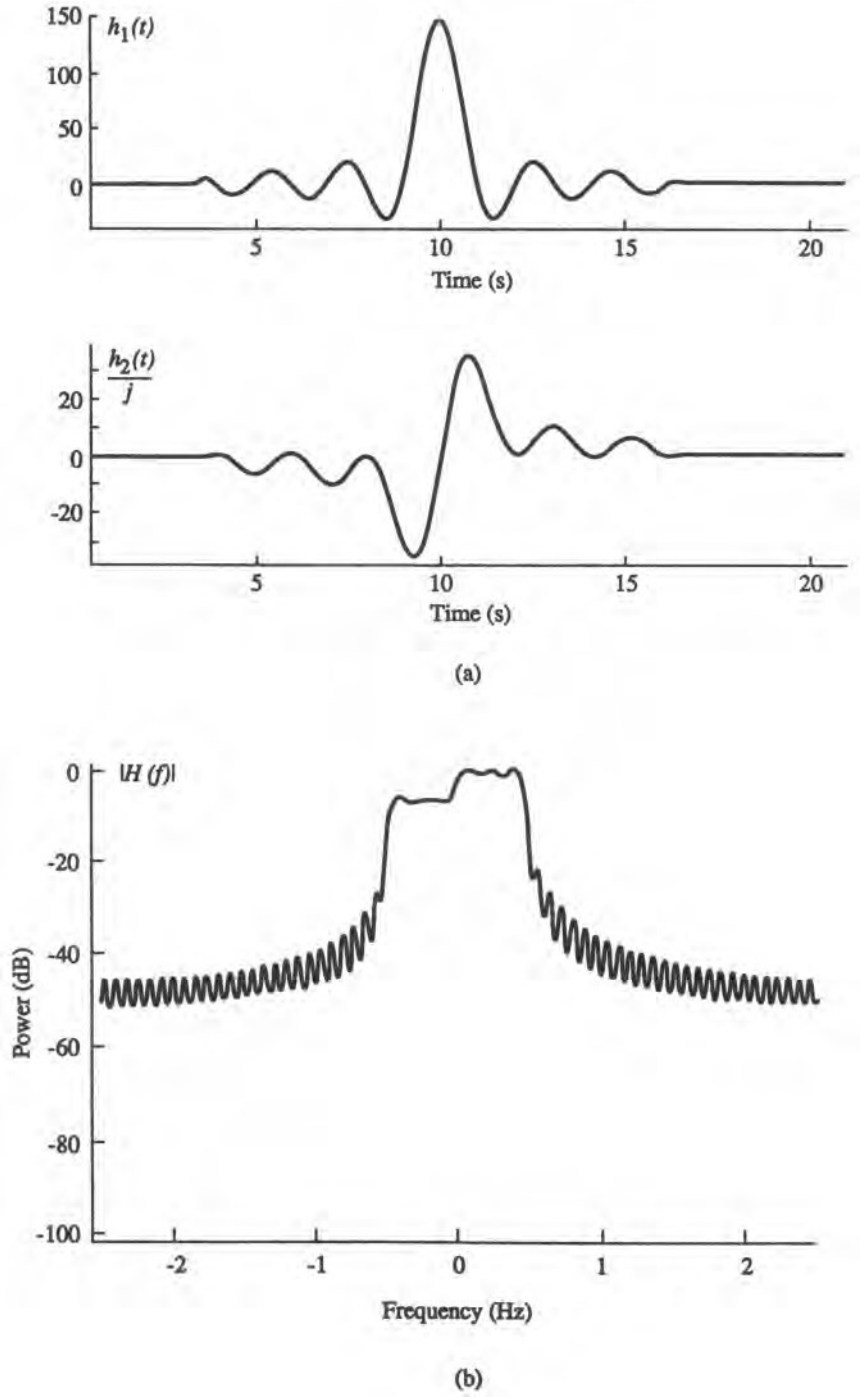
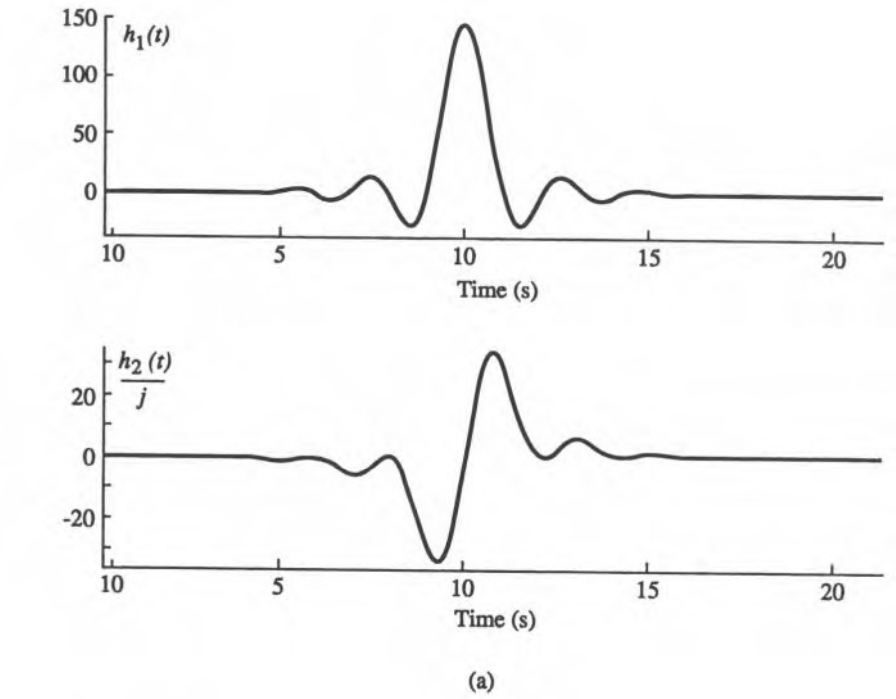
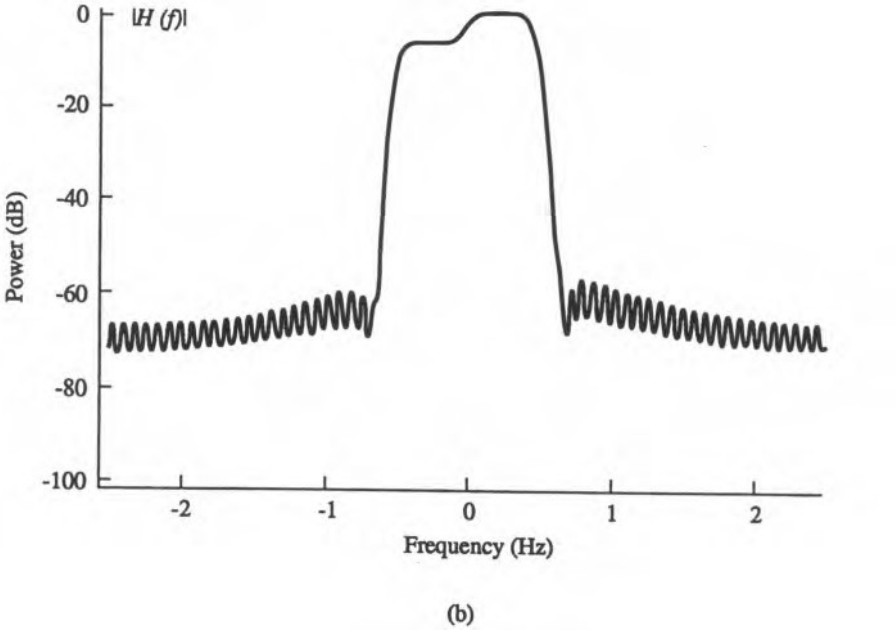


Figure 4.18. (a) Truncated real and imaginary components of the lowpass-equivalent impulse response $h(t)$; (b) lowpass-equivalent frequency response of filter with truncated impulse response.



(a)



(b)

Figure 4.19. (a) Windowed real and imaginary components of the lowpass-equivalent impulse response $h(t)$; (b) lowpass-equivalent frequency response of filter with truncated impulse response.

generally be a larger number of points within the truncated response. Here, FFT-type filtering is generally preferred because of computational efficiency. The disadvantages are:

1. For nonperiodic processes we have to use the overlap-and-add or overlap-and-save methods. This adds some complexity to the simulation.
2. FFT filtering is inherently a block process. Input signal samples are accumulated in arrays before each filter process, thus creating artificial delays within the simulated system. This adds complexity when combining signals within a simulated system. It also prevents using these filters within a feedback loop.

The FIR filter techniques described above are used for simulation of analog filters in communication systems. However, as noted before, FIR filters are used extensively in digital signal processing. These frequency-selective filters are designed directly in the discrete-time domain. The techniques used in the design of these filters are amply described in the literature.[1,2] The two most frequently used methods in frequency-selective filter design are the following:

1. *The windowing design method*, which is similar to the one described above for simulation.
2. *Equiripple approximation in the frequency domain for FIR filters.* Several techniques have been developed to obtain an optimal Chebyshev solution to the equiripple FIR filter. The most commonly used is the algorithm described by Parks and McClellan.[12]

4.1.3. Simulation of Filtering with IIR Filters

Infinite impulse responses in continuous systems can arise in a number of ways, but are most often associated with systems described by linear time-invariant differential equations (LTI/DE), such as the "classical" filters inroduced earlier. The natural discrete counterparts to such continuous systems are LTI difference equations. We saw earlier, in fact, that one type of s-to-z mapping transforms a continuous LTI/DE system into one described by an LTI difference equation. We also pointed out in conjunction with (4.1.22) that direct computation of the convolution sum is not feasible for an IIR filter. An equivalent formulation for the output of an IIR system that *is* computable is in fact a (recursive) difference equation. Thus, filtering with IIR filters can be implemented in simulation with recursive structures.

We thus begin this section with a very brief introduction to linear difference equations and in particular their z-domain representations. We then present the standard structural interpretations through which such discrete systems may be realized. We then look at two often-used s-to-z mappings through which continuous structures may be transformed into discrete ones. We then summarize the steps for producing IIR discrete systems from continuous ones, and illustrate the process with examples.

4.1.3.1. Systems Characterized by Linear Constant-Coefficient Difference Equations

Let us consider the case of a general Nth-order difference equation,

$$\sum_{k=0}^{N} b_k y(n-k) = \sum_{k=0}^{M} a_k x(n-k) \tag{4.1.49a}$$

Suppose $\{x(n)\}$ is the input sequence and $\{y(n)\}$ the output sequence. Since we can write (4.1.49a) as (with the usual normalization $b_0 = 1$)

$$y(n) = \sum_{k=0}^{M} a_k x(n-k) - \sum_{k=1}^{N} b_k y(n-k) \tag{4.1.49b}$$

we can see that the output at any time depends not only on the present and past inputs, but also on previous outputs. This latter fact is what makes the system *recursive*. In the case $b_k = 0$, for all k, (4.1.49b) reduces to the FIR filtering model discussed above.

Applying the z-transform to both sides of the equation and using the linearity and time-shifting property yields

$$\sum_{k=0}^{N} b_k z^{-k} Y(z) = \sum_{k=0}^{M} a_k z^{-k} X(z)$$

or

$$Y(z) \cdot \sum_{k=0}^{N} b_k z^{-k} = X(z) \sum_{k=0}^{M} a_k z^{-k}$$

so that

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} a_k z^{-k}}{\sum_{k=0}^{N} b_k z^{-k}} \tag{4.1.50}$$

Thus, the transfer function of a linear constant-coefficient difference equation is *always* rational.

The zeros of $H(z)$ are specified by the roots of the numerator and the poles by the roots of the denominator of (4.1.50).

Since the difference equation (4.1.49) has real coefficients, the poles and zeros of $H(z)$ are either real or occur in complex conjugate pairs.

The complex conjugate roots may be combined to produce biquadratic polynomials

$$(1 + cz^{-1})(1 + c^* z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2}$$

where $a_1 = c + c^*$ and $a_2 = c \cdot c^*$ are real numbers. Thus, the transfer function $H(z)$ can be represented as a product of M_1 real roots and M_2 biquadratic factors. All coefficients in the rational function $H(z)$ are real,

$$H(z) = K \frac{\prod_{i=0}^{M_1} (1 + a_i z^{-i}) \prod_{i=0}^{M_2} (1 + b_{1i} z^{-1} + b_{2i} z^{-2})}{\prod_{i=0}^{N_1} (1 + c_i z^{-1}) \prod_{i=0}^{N_2} (1 + d_{1i} z^{-1} + d_{2i} z^{-2})} \tag{4.1.51}$$

where $M = M_1 + 2M_2$ and $N = N_1 + 2N_2$.

4.1.3.2. Structures of Recursive Discrete Filters Implemented in Simulation Models

The widespread use of *digital signal-processors* (DSP) has prompted the development of many methods of realization of discrete filters.[1–4, 13] Two of these methods are of particular interest in communication system simulation and are described below. The development of this section parallels that in the corresponding s-domain case that was examined in Section 3.6.6.1.

4.1.3.2.1. Direct-Form (Canonic) Realization. The result in (4.1.50) can, without loss of generality, be rewritten in a form where we set the order of both polynomials in the

numerator and denominator to the larger value, say M. If $M \neq N$, some of the coefficients in the lower order polynomial can simply be set to zero. Thus, the transfer function $H(z)$ can be represented as a cascade connection of $H_1(z)$ and $H_2(z)$,

$$H(z) = H_1(z) \cdot H_2(z) = \frac{W(z)}{X(z)} \cdot \frac{Y(z)}{W(z)}$$

$$= \left[\frac{1}{\sum_{k=0}^{M} b_k z^{-k}}\right]\left[\sum_{k=0}^{M} a_k z^{-k}\right] \tag{4.1.52}$$

The function $W(z)$ introduced in (4.1.52) is an artifice that facilitates the development of an explicit realization for $H(z)$.

Since cascade connections are commutative [see (3.2.5)], we can arbitrarily assign the transfer functions $W(z)/X(z)$ or $Y(z)/W(z)$ to $H_1(z)$ or $H_2(z)$.

A simpler realization of the discrete transfer function results from the assignment

$$H_1(z) = \frac{W(z)}{X(z)} = \frac{1}{\sum_{k=0}^{M} b_k z^{-k}} \tag{4.1.53a}$$

and

$$H_2(z) = \frac{Y(z)}{W(z)} = \sum_{k=0}^{M} a_k z^{-k} \tag{4.1.53b}$$

The expression for $H_1(z)$ in (4.1.53a) can be rewritten in the form

$$X(z) = \sum_{k=0}^{M} b_k W(z) z^{-k}$$

which can be rephrased as

$$W(z) = \frac{1}{b_0}\left[X(z) - \sum_{k=1}^{M} b_k W(z) z^{-k}\right] \tag{4.1.54}$$

The difference equation corresponding to (4.1.54) is

$$w(n) = \frac{1}{b_0}\left[x(n) - \sum_{k=1}^{M} b_k w(n-k)\right]$$

The expression for $H_2(z)$ in (4.1.53b) can be rewritten in the form

$$Y(z) = \sum_{k=0}^{M} a_k W(z) z^{-k} \tag{4.1.55}$$

The difference equation corresponding to (4.1.55) is

$$y(n) = \sum_{k=0}^{M} a_k w(n-k)$$

Upon examining (4.1.54) and (4.1.55), we note that both equations contain chains of delays with identical inputs $W(z)$. Therefore, the corresponding transfer functions can be realized by a single chain of delays as shown in Figure 4.20. In this configuration, implementation of the difference equation requires only M delay elements. This realization is usually referred to as the *canonic realization* because it requires the minimum number of delay elements for implementation of (4.1.50).

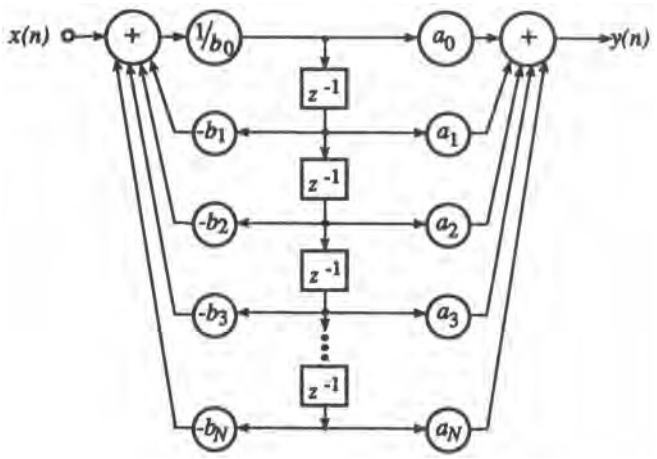


Figure 4.20. Discrete recursive filter, canonic realization.

4.1.3.2.2. The Cascade Interconnections of Biquadratic Canonic Sections. The canonic structure in Figure 4.20 was obtained directly from the transfer function $H(z)$ in (4.1.50). We can also realize the transfer function from the factored form of $H(z)$ in (4.1.52).

It is convenient to think of $H(z)$ in general form as a cascade connection of biquadratic sections, as was done in the s-domain case in Section 3.6.6.1,

$$H(z) = K \prod_{i=0}^{L} \frac{1 + \alpha_{1i} z^{-1} + \alpha_{2i} z^{-2}}{1 + \beta_{1i} z^{-1} + \beta_{2i} z^{-2}} \tag{4.1.56}$$

where L is $(M + 1)/2$. In this form we assumed that real poles or real zeros were combined in pairs. If there is an odd number of real poles or zeros, one of the coefficients α_{2i} or β_{2i} will simply be zero.

A cascade realization of a fourth-order system is shown in Figure 4.21. Each section in Figure 4.21 is the z-domain counterpart of the s-domain realization previously shown in Figure 3.27c. The cascade realization is the most commonly used model in communication systems simulation and in implementation because of its modular form. That is, since an arbitrary filter can be represented by a cascade of biquadratic sections, one needs to develop only a single, fairly simple program to simulate a biquadratic section (with coefficient values as inputs), and to simulate a more complicated filter, one simply makes multiple calls to the same basic program.

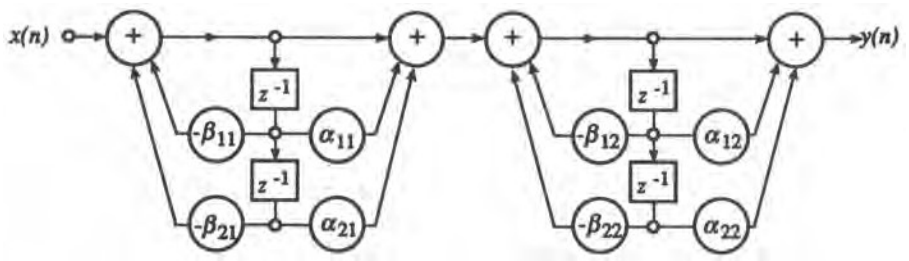


Figure 4.21. Cascade structure with a canonic realization for each second-order subsystem (biquad).

4.1.3.2.3. The Parallel Realization. The parallel-form discrete filter structure is obtained by performing a partial fraction expansion of the transfer function $H(z)$ in (4.1.50),

$$H(z) = \frac{b_n}{a_n} + \sum_{k=1}^{N} \frac{A_k}{1 + B_k z^{-1}} \tag{4.1.57}$$

Combining the complex conjugate poles into biquadratic forms yields

$$H(z) = C + \sum_{i=1}^{N/2} \frac{\alpha_{0i} + \alpha_{1i} z^{-1}}{1 + \beta_{1i} z^{-1} + \beta_{2i} z^{-2}} \tag{4.1.58}$$

Any two real poles can also be combined to yield a biquadratic section. If there is an odd number of real poles, the appropriate α_1 and β_2 are simply set to zero.

A parallel-form realization for a fourth-order filter using canonic biquadratic sections is shown in Figure 4.22.

4.1.3.3. Transformations between Continuous-Time and Discrete-Time Systems Represented by Rational Functions

In this section we consider two transformations often used for mapping continuous-time systems with rational transfer functions (hence, infinite impulse responses) to discrete-time systems with rational transfer functions. The first is a technique referred to as *impulse-invariant transformation* and is used mainly in modeling filters specified by the impulse response in the time domain. In this technique a continuous system is transformed into a discrete system by setting the impulse response of the latter to a sampled version of the impulse response of the continuous system. The discrete impulse response is obtained by sampling the continuous impulse response at a rate sufficient to avoid severe aliasing (see Section 3.5 on the sampling theorem).

The second technique we discuss is the *bilinear transformation*, which is a particular mapping from the s domain to the z domain belonging to the general class (4.1.32). This

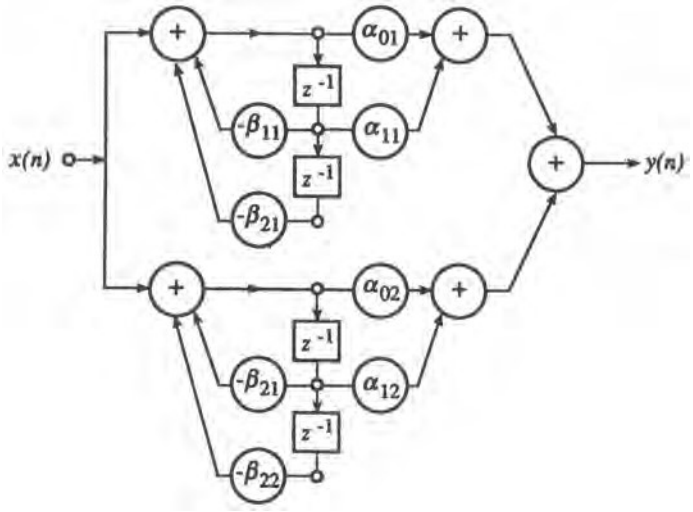


Figure 4.22. Parallel-form structure with a canonic realization for each second-order subsystem (biquad) associated with the bilinear transformation.

transform is used mainly for filters specified by masks in the frequency domain which are attempted to be met by one or another classical filter. It is therefore the preferred mapping method used in simulation of continuous systems except when specific time-domain characteristics are of paramount importance.

4.1.3.3.1. Impulse-Invariant Transformation. The characteristic property of the impulse-invariant transformation is that the impulse response of the resultant discrete filter is a sampled version of the impulse response of the continuous filter. As a consequence, the frequency response of the discrete filter is an aliased version of the continuous filter.

In (3.7.1)–(3.7.5) it was shown that, to map a continuous function $h_c(t)$ with a transfer function $H_c(s)$ into a discrete function $h_d(nT_s)$ with a transfer function $H_d(z)$, one can use the mapping

$$z = e^{sT_s} \tag{4.1.59}$$

It can be shown as a generalization of the sampling theorem in (3.5.6) that the z-transform is related to the Laplace transform of $h_c(t)$ by the equation

$$H_d(z)|_{z=\exp(sT_s)} = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} H_c\left[s + j\frac{2\pi}{T_s}k\right] \tag{4.1.60}$$

From the above relationship it is seen that strips of width $2\pi/T_s$ in the s-plane are mapped into the entire z-plane as shown in Figure 4.23.

The left half of each strip maps into the interior of the unit circle and the right half of the strip maps into the exterior of the unit circle. Each $2\pi/T_s$ segment of the imaginary axis maps linearly once around the unit circle. Therefore the frequency response of the discrete filter is evaluated for $z = e^{sT_s}|_{s=j\omega}$ and is given by

$$H_d(e^{j\omega T_s}) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} H_c\left[j\omega + j\frac{2\pi}{T_s}k\right] \tag{4.1.61}$$

From the above relationship, it is clear that the impulse-invariant transformation can introduce severe aliasing if the frequency response of the continuous filter spans more than one strip. Thus, to avoid aliasing, we have

$$H_c(f) = 0 \qquad \text{for} \quad |f| \geq f_s/2 \tag{4.1.62}$$

where $\omega = 2\pi f$ and $f_s = T_s^{-1}$. Equation (4.1.62) is merely a reiteration of the sampling theorem in Section 3.5.

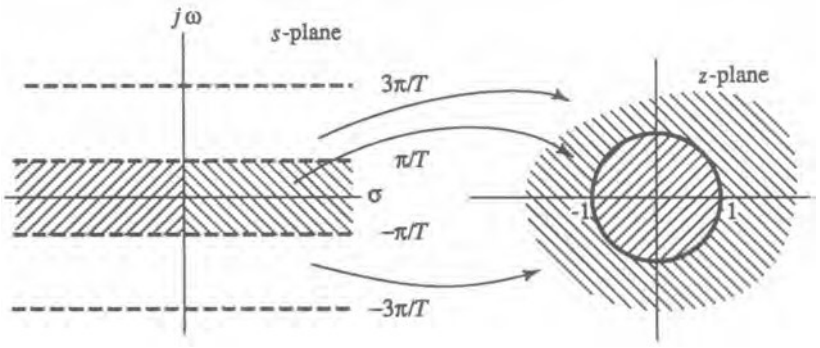


Figure 4.23. The mapping from the s plane to the z plane defined by $z = e^{sT}$.

Impulse-Invariant Mapping for Rational Functions. To demonstrate how one implements the impulse-invariant mapping technique, assume that the continuous filter has a transfer function such that the partial fraction expansion has the form

$$H_c(s) = \sum_{i=1}^{m} \frac{A_i}{s - s_i} \tag{4.1.63}$$

The impulse response of this filter is

$$h_c(t) = \sum_{i=1}^{m} A_i e^{s_i t} u(t) \tag{4.1.64}$$

The z-transform of the impulse response is

$$H_d(z) = \sum_{n=0}^{\infty} h_c(nT_s) z^{-n} \tag{4.1.65}$$

which, upon substitution of (4.1.64), gives

$$H_d(z) = \sum_{n=0}^{\infty} \sum_{i=1}^{m} A_i (e^{s_i T_s} z^{-1})^n$$

and which, after summing on n, yields

$$H(z) = \sum_{i=1}^{m} \frac{A_i}{1 - e^{s_i T_s} z^{-1}} \tag{4.1.66}$$

By comparing (4.1.63) to (4.1.66), it is seen that simple poles are mapped using the relation

$$\frac{1}{s - s_i} \rightarrow \frac{1}{1 - z^{-1} e^{s_i T_s}} \tag{4.1.67a}$$

When s_i is complex the poles appear in complex conjugate pairs

$$\frac{a_i}{s - s_i} + \frac{a_i^*}{s - s_i^*} = \frac{2\gamma_i s - 2(\sigma_i \gamma_i + \omega_i \delta_i)}{s^2 - 2\sigma_i s + (\sigma_i^2 + \omega_i^2)} \tag{4.1.67b}$$

where $s_i = \sigma_i + j\omega_i$ and $a_i = \gamma_i + j\delta_i$.

By substituting (4.1.67a) into each of the terms on the left side of (4.1.67b), we get

$$\frac{a_i}{1 - z^{-1} e^{s_i T_s}} + \frac{a_i^*}{1 - z^{-1} e^{s_i^* T_s}}$$

$$= \frac{2\gamma_i - z^{-1} e^{\sigma_i T_s} [2\gamma_i \cos(\omega_i T_s) + 2\delta_i \sin(\omega_i T_s)]}{1 - z^{-1} 2e^{\sigma_i T_s} \cos(\omega_i T_s) + z^{-2} e^{2\sigma_i T_s}}$$

Hence a biquadratic term in the s domain is mapped into a biquadratic term in the z domain as

$$\frac{\gamma_i s - (\sigma_i \gamma_i + \omega_i \delta_i)}{s^2 - 2\sigma_i s + (\sigma_i^2 + \omega_i^2)}$$

$$\rightarrow 2 \frac{\gamma_i - z^{-1} e^{\sigma_i T_s} [\gamma_i \cos(\omega_i T_s) + \delta_i \sin(\omega_i T_s)]}{1 - z^{-1} 2e^{\sigma_i T_s} \cos(\omega_i T_s) + z^{-2} e^{2\sigma_i T_s}} \tag{4.1.68}$$

From (3.6.23) and (4.1.68) we find that a biquadratic partial-fraction-expanded rational function in the s domain

$$H(s) = \sum_{i=1}^{n} \frac{s - \sigma_i - \omega_i(\delta_i/\gamma_i)}{s^2 - 2\sigma_i s + \sigma_i^2 + \omega_i^2} \tag{4.1.69}$$

is mapped into a partial-fraction-expanded rational function in the z domain

$$H(z) = \sum_{i=1}^{n} \frac{1 - z^{-1}e^{\sigma_i T_s}[\cos(\omega_i T_s) + (\delta_i/\gamma_i)\ \sin(\omega_i T_s)]}{1 + 2z^{-1}e^{\sigma_i T_s}\ \cos(\omega_i T_s) + z^{-2}e^{2\sigma_i T_s}} \tag{4.1.70}$$

where $n = m/2$. It should be noted that this modeling technique yields the parallel realization in Figure 4.22 with $\alpha_{0i} = 1$, $\alpha_{1i} = -e^{\sigma_i T_s}[\cos(\omega_i T_s) + (\sigma_i/\gamma_i)\sin(\omega_i T_s)]$, $\beta_{1i} = 2e^{\sigma_i T_s}\cos(\omega_i T_s)$, and $\beta_{2i} = e^{2\sigma_i T_s}$.

In summary, the use of impulse invariance preserves the time-domain behavior of the filter. To avoid aliasing, the frequency response of the continuous filter must be limited to the band $-f_s/2 \le f \le f_s/2$. In some situations, the incoming signal may have high-frequency components that are not of interest, or the actual filter bandwidth may be much larger than that of the signal itself, as in a highpass filter. In these cases, in order to avoid aliasing distortion, one can use a suitable lowpass filter (sometimes called a prefilter) to produce a bandlimited response that will not suffer from aliasing at the chosen sampling rate.

A summary of the procedure for transforming a filter defined by its poles and zeros into an impulse-invariant discrete equivalent is shown in the flowchart of Figure 4.24. As the first step, if applicable, this filter should be transformed into a lowpass equivalent. The continuous filter is next transformed into the discrete form. In the example below, we transform the lowpass equivalent of a filter into a discrete filter ready to be used in a simulation.

■ *Example 4.1.7.* Consider

$$H(s) = \frac{1}{(s+2)(s+3)} = \frac{1}{s+2} - \frac{1}{s+3}$$

The squared amplitude of the frequency response of this filter is

$$|H([j\omega)|^2 = \frac{1}{(6-\omega^2)^2 + 25\omega^2}$$

At the frequency $f_c = \omega_c/(2\pi) = 12.5$ Hz the frequency response is down to –60 dB, which, for this example, defines the bandwidth:

$$\frac{|H([j\omega)|^2_{\omega_c = 2\pi 12.5}}{|H([j\omega)|^2_{\omega=0}} = -60 \text{ dB}$$

We therefore choose the sampling rate $f_s = 2f_c = T_s^{-1} = 25$ samples/s. The discrete filter, using the mapping in (4.1.67a), is then

$$H(z) = \left(\frac{1}{1 - z^{-1}e^{-2T_s}} - \frac{1}{1 - z^{-1}e^{-3T_s}}\right)$$

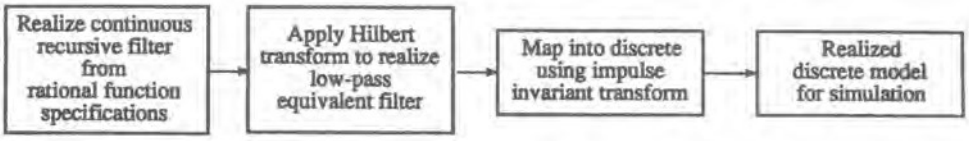


Figure 4.24. Flowchart for modeling of recursive filters from rational functions for simulation.

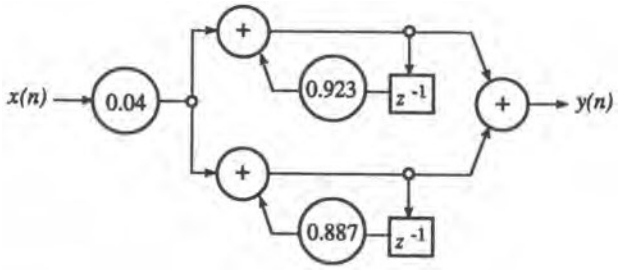


Figure 4.25. Parallel realization of filter in Example 4.1.7.

The parallel realization of the discrete filter above is shown in Figure 4.25.

The frequency response is evaluated from (3.7.12), which results in

$$H(\omega) = T_s|H(z)|_{z=\exp(j\omega T_s)} = T_s\left|\frac{e^{-j\omega T_s}(e^{-2T_s} - e^{-3T_s})}{1 - e^{-j\omega T_s}(e^{-2T_s} + e^{-3T_s}) + e^{-j2\omega T_s}e^{-5T_s}}\right|$$

The frequency responses for amplitude and phase of the continuous and discrete filters at various sampling rates are shown in Figures 4.26a and 4.26b. It is clearly seen from the plots that there can be considerable aliasing present. We should therefore use extreme caution when using the impulse-invariant transform. ■

4.1.3.3.2. The Bilinear Transformation. In Section 4.1.1.6 we introduced a general form of s-to-z mappings

$$\frac{1}{s} = f(z^{-1})$$

which we developed for the case where f is a ratio of polynomials in z^{-1}. By far the most commonly used instance of this type of mapping is the so-called bilinear transformation,

$$s = \frac{2}{T_s}\frac{1 - z^{-1}}{1 + z^{-1}} \tag{4.1.71a}$$

or

$$\frac{1}{s} = \frac{T_s}{2}\frac{1 + z^{-1}}{1 - z^{-1}} \tag{4.1.71b}$$

which has many desirable properties. To obtain the digital structure corresponding to a continuous filter $H(s)$, we use

$$H_d(z) = H\left(\frac{2}{T_s}\frac{1 - z^{-1}}{1 + z^{-1}}\right) \tag{4.1.72}$$

The order of the denominator of $H_d(z)$ is the same as that of $H(s)$; however, the order of the zeros may differ.

■ *Example 4.1.8.* Consider

$$H(s) = \frac{1}{s + a}; \qquad H_d(z) = \frac{1 + z^{-1}}{2/T_s + a + z^{-1}(a - 2/T_s)}$$

The pole at $s = -a$ is mapped to a pole at $(2/T_s - a)/(2/T_s + a)$ in the z domain and the zero at ∞ in s is sent to -1 in the z domain. ■

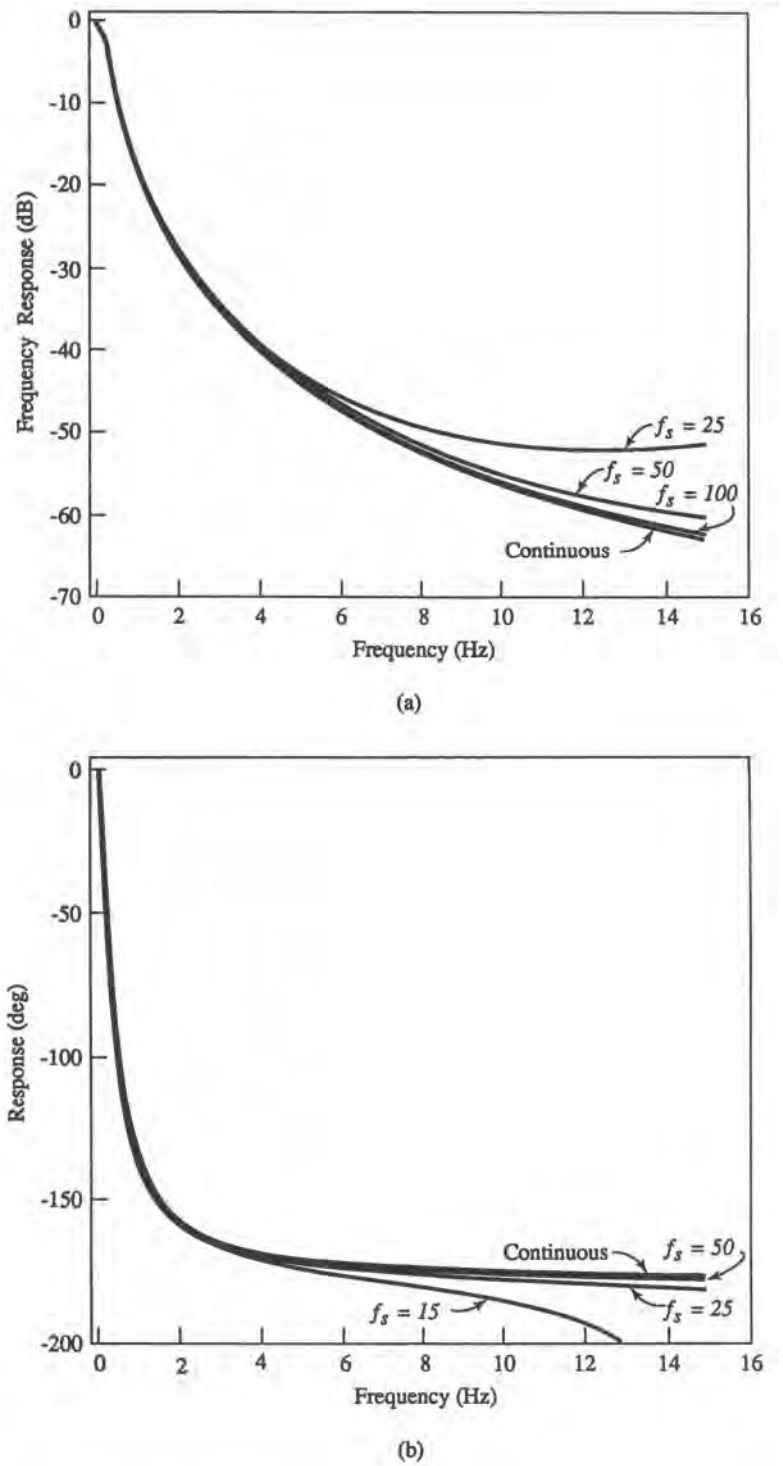


Figure 4.26. (a) Continuous and discrete frequency responses for different sampling rates. (b) Continuous and discrete phase responses for different sampling rates.

Note that the substitution

$$\frac{1}{s} \rightarrow \frac{T_s}{2}\frac{1+z^{-1}}{1-z^{-1}}$$

corresponds to the *trapezoidal rule* integration formula, which is a second-order implicit integration formula given by the difference equation (see Section 5.4)

$$y_n = y_{n-1} + \frac{T_s}{2}(x_n + x_{n-1})$$

By taking the z-transform of the above difference equation, we get the transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{T_s}{2}\frac{1+z^{-1}}{1-z^{-1}}$$

which, of course, is identical to the expression (4.1.71b).

This transformation is invertible. The inverse mapping is given by

$$z = \frac{2f_s + s}{2f_s - s} \tag{4.1.73}$$

where $f_s = T_s^{-1}$. If we let $s = \sigma + j\omega$, we get

$$z = \frac{2f_s + \sigma + j\omega}{2f_s - \sigma - j\omega}$$

For $\sigma < 0$ we find $|z| < 1$ and for $\sigma > 0$, $|z| > 1$. The entire left half-plane is mapped in the interior of the unit circle (see Figure 4.27) and the right half-plane is mapped to the outside of the unit circle.

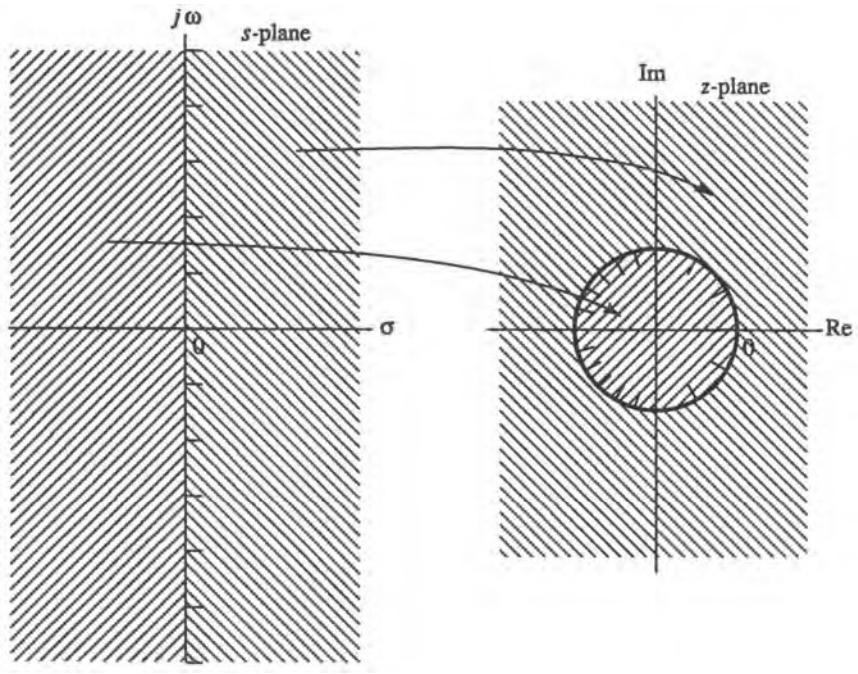


Figure 4.27. Mapping from the s plane to the z plane for the bilinear transformation.

Since the entire $j\omega$ axis of the s plane is mapped onto the unit circle in the z plane, aliasing errors are eliminated. There is, however, a nonlinear relationship between the continuous frequency ω and the discrete frequency Ω. Indeed, for $s = j\omega$ and $z = e^{j\Omega T_s}$ we get from (4.1.7la)

$$j\omega \rightarrow \frac{2}{T_s}\frac{1 - e^{-j\Omega T_s}}{1 + e^{-j\Omega T_s}}$$

or

$$\omega \rightarrow \frac{2}{T_s}\tan\left(\frac{\Omega T_s}{2}\right) \tag{4.1.74}$$

For small values of Ω, the mapping is almost linear. For most of the frequency scale it is highly nonlinear. Thus, the bilinear transform preserves the amplitude, but warps the frequency (see Figure 4.28). This implies that the bilinear transform will satisfy specifications only when the frequency response is described by a mask composed of piecewise-constant bands, and the exact shape of the response within the bands is of no consequence (see Figure 4.2). This requirement is met by a large class of filters which includes all classical filters described earlier. These filters can be modified from their original definition so that the frequency warping is compensated. This prewarping is performed at a set of critical frequencies and is best illustrated by example.

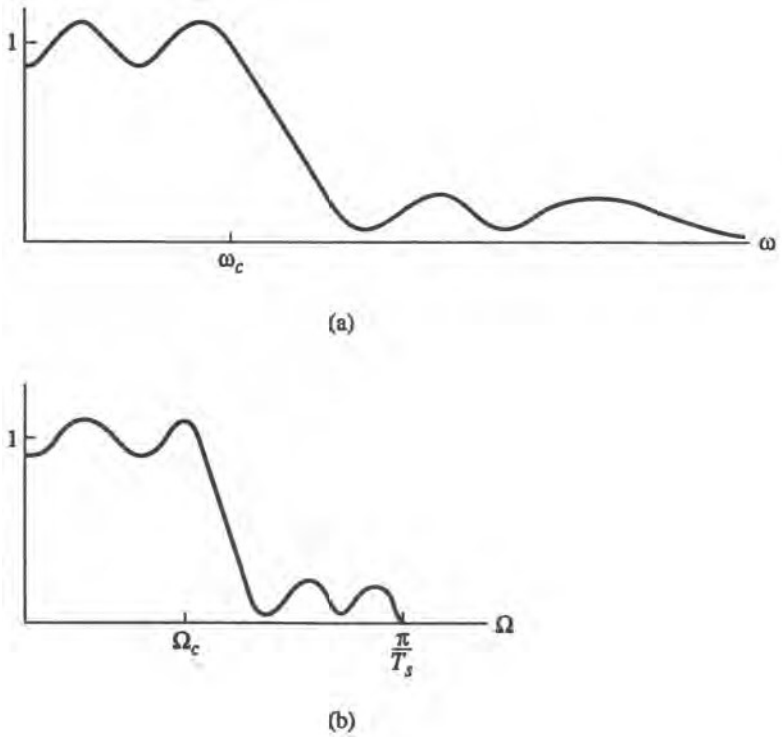


Figure 4.28. Nonlinear frequency warping caused by the bilinear transformation. (a) Transfer function in continuous domain; (b) transfer function in discrete domain.

■ *Example 4.1.9.* Figure 4.29 shows the frequency response of a bandpass filter. The set of desired cutoff frequencies of the discrete filter $(\Omega_1, \Omega_2, \Omega_3, \Omega_4)$ is shown below the abscissa of Figure 4.29. Using the frequency warping relation (4.1.74), we find the new set of cutoff frequencies of the continuous filter $(\omega_1, \omega_2, \omega_3, \omega_4)$. The continuous filter with the new set of cutoff frequencies is shown on the ordinate of Figure 4.29. ■

The advantages and disadvantages of the bilinear mapping are summarized below.

Advantages of the bilinear transformation:

1. Mapping the s plane to the z plane is one-to-one.
2. Aliasing is avoided.
3. The closed left half of the s plane is mapped onto the unit disk of the z plane.

Disadvantages of the bilinear transformation:

1. The frequency scale is warped.
2. Waveforms are not preserved if T_s is not small enough.

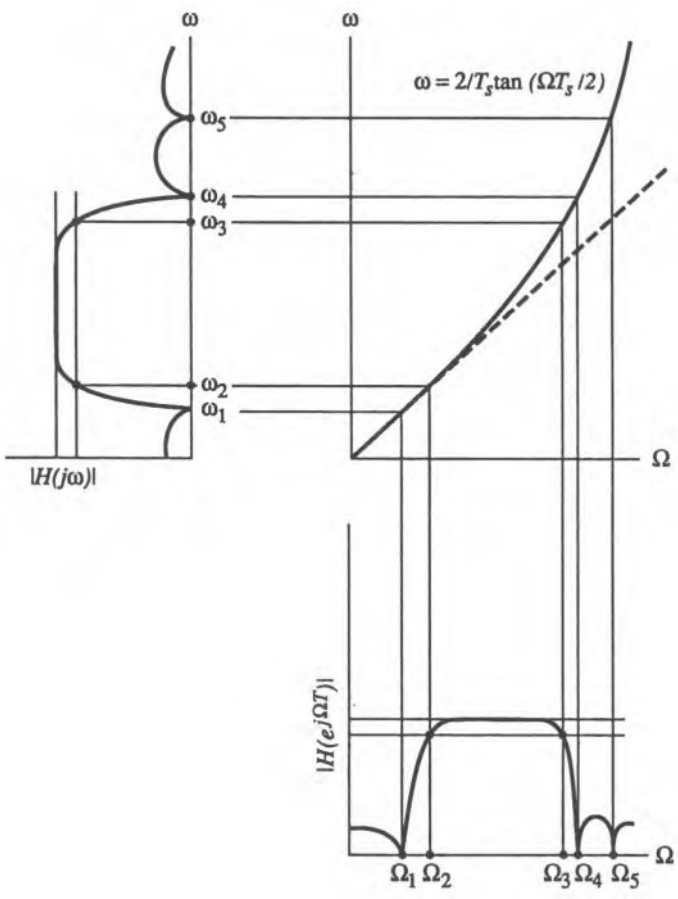


Figure 4.29. Technique for compensating the nonlinear frequency warping of the bilinear transformation.

Most classical filters can be modeled as a cascade connection of biquadratic sections. The coefficients of these sections are computed from the filter specifications including the prewarped cutoff frequencies. The biquadratic sections for the continuous filter are transformed into biquadratic sections of the discrete filter using the mapping

$$\frac{\alpha_1 s^2 + \beta_1 s + \gamma_1}{\alpha_2 s^2 + \beta_2 s + \gamma_2} \rightarrow \frac{z^{-2}(4f_s^2\alpha_1 - 2f_s\beta_1 + \gamma_1) + z^{-1}(2\gamma_1 - 8f_s^2\alpha_1) + (4f_s^2\alpha_1 + 2f_s\beta_1 + \gamma_1)}{z^{-2}(4f_s^2\alpha_2 - 2f_s\beta_2 + \gamma_2) + z^{-1}(2\gamma_2 - 8f_s^2\alpha_2) + (4f_s^2\alpha_2 + 2f_s\beta_2 + \gamma_2)} \tag{4.1.75}$$

This modeling technique yields a cascade realization as in Figure 4.21.

In summary, the bilinear transformation provides a simple mapping between continuous and discrete filters which is free of aliasing. It is ideal for filters in the frequency domain that have a piecewise-constant frequency response specification. One disadvantage of this technique is that neither the impulse response nor the phase response of the continuous filter is preserved.

4.1.3.3.3. Effect of Mapping on Lowpass-Equivalent Filters Represented by Rational Functions. Lowpass equivalents of narrowband bandpass filters represented by rational functions were derived in Section 4.1.1.4. The procedure was based on the reasoning that for narrowband bandpass filters ($B \ll f_0$) the positive-frequency response is dominated by the poles and zeros located in the positive-frequency half of the s plane. For narrowband bandpass filters with $B \ll f_0$ the poles and zeros are concentrated about the center frequencies $\pm f_0$ as in Figure 4.7a. To obtain the lowpass equivalent we substitute $f \rightarrow f + f_0$. The contribution of the negative-frequency poles and zeros, located now about $-2f_0$, to the lowpass frequency response is negligible for continuous filters. However, if these filters are mapped into the z domain, this contribution may be substantial and cannot be neglected.

If the continuous lowpass equivalents of bandpass filters are mapped into discrete filters via the *impulse-invariant transformation*, it is necessary to discard the negative-frequency poles located about $-2f_0$, since they may fold into the lowpass region and introduce severe aliasing. That is, we cannot merely translate the bandpass characteristics so that the response about $+f_0$ is now centered about $f = 0$, but we must also filter out anything else. This filtering is accomplished, as described in Section 4.1.1.4, by discarding the poles corresponding to the response about $-2f_0$.

The *bilinear transformation* does not introduce aliasing distortion. However, because of nonlinear frequency warping, the frequency band at $-2f_0$ is moved closer to the lowpass response, which will increase distortion.

4.1.3.3.4. Guide for Mapping Recursive Filters Specified in Frequency Domain. Filters that are specified by a mask of piecewise-constant acceptance strips as shown in Figure 4.2 are usually realized by classical filters. These filters are characterized by the *type*, for example third-order Butterworth or fifth-order elliptic, and by a small number of parameters at a few critical frequencies. The flowchart for preparation of classical filters for simulation is shown in Figure 4.30. The steps in the flowchart are briefly described below.

1. Select a lowpass-equivalent filter for bandpass or band-reject filters. If the center frequency is much larger than the bandwidth, then $\omega_0 >> \omega_b$ (Section 4.1.1.4). In that case the lowpass-equivalent filter is identical to a lowpass prototype filter possibly shifted in frequency.

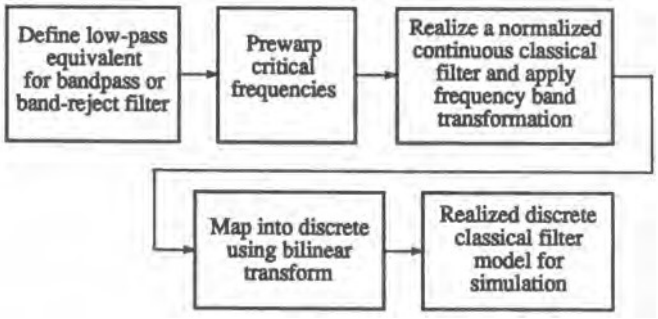


Figure 4.30. Flowchart for modeling of classical filters for simulation.

2. Since the bilinear transformation is most often used for mapping these filters, we have to prewarp the critical frequencies using the relationship in (4.1.74).
3. Next, the filter type and parameters are used to obtain, from filter design tables, the coefficients of the rational functions (usually in biquadratic form) for the normalized lowpass realization of the filter. The filter is then transformed into the appropriate continuous filter using the prewarped critical frequencies. The transformation to obtain the actual filter parameters is done using the relations in Section 4.1.1.3.
4. The biquadratic sections of the continuous filter in the s domain are transformed into the discrete biquadratic sections of the z-domain filter using the relation (4.1.75).

■ *Example 4.1.10.* To illustrate the application of the above procedure, let us simulate a lowpass filter using a Chebyshev filter type with the following specifications:

1. Passband ripple $\alpha_p = 2\,\text{dB}(\epsilon = 0.7648)$
2. Stopband loss $\alpha_r = 22\,\text{dB}$
3. Passband edge frequency $F_p = \Omega_p/(2\pi) = 100\text{Hz}$
4. Frequency where the prescribed stopband loss is achieved $F_r = \Omega_r/(2\pi) = 200\,\text{Hz}$
5. Sampling rate $f_s = 1/T_s = 600\ \text{samples/s}$

From the prewarping relation in (4.1.74) we compute that the prewarped frequencies of the continuous filter $f_p = 110.26\,\text{Hz}$ and $f_r = 330.79\,\text{Hz}$. The transition frequency ratio is then $f_r/f_p = 3$. The filter order that satisfies the above specifications is computed from (4.1.7) and results in $n = 2$. The transfer function of a second-order normalized lowpass Chebyshev filter with $\alpha_p = 2\,\text{dB}$ is, from Table 4.A.3 in the Appendix,

$$H(s) = \frac{1}{s^2 + 0.804s + 0.637}$$

Using the lowpass-to-lowpass transformation in (4.1.13), with $\omega_b = 2\pi f_p = 692.92$, we get the prewarped continuous filter

$$H(s) = \frac{4.77 \times 10^5}{(s^2 + 555s + 3.04 \times 10^5)}$$

By applying the bilinear transformation of (4.1.7la), we obtain the transfer function of the discrete filter

$$H(z) = \frac{(z^{-1} + 1)^2}{2.26z^{-2} - 4.75z^{-1} + 5.05} = 0.198\frac{z^{-2} + 2z^{-1} + 1}{0.436z^{-2} - 0.94z^{-1} + 1.0} \qquad (4.1.76)$$

The frequency response of the discrete filter is

$$|H(z)|^2_{z=\exp(j\Omega T_s)} = \left| \frac{(e^{-j\Omega T_s} + 1)^2}{2.26e^{-j2\Omega T_s} - 4.75e^{-j\Omega T_s} + 5.05} \right|^2$$

The canonic realization of the discrete filter in (4.57) is shown in Figure 4.31a, and the frequency response in Figure 4.31b.

It should be noted that the above filter is a discrete model for a lowpass filter with specified passband and stopband tolerances. It is *not* a simulation of a continuous second-order Chebyshev filter. Indeed, a design of a continuous filter with the same requirements will produce a third-order Chebyshev filter. To simulate the performance of a given filter design, we will have to resort to a much higher sampling rate, to reduce frequency warping. For example, to obtain a warping effect of no more than 2% at a given frequency f, the sampling rate has to be $f_s = 13f$. ■

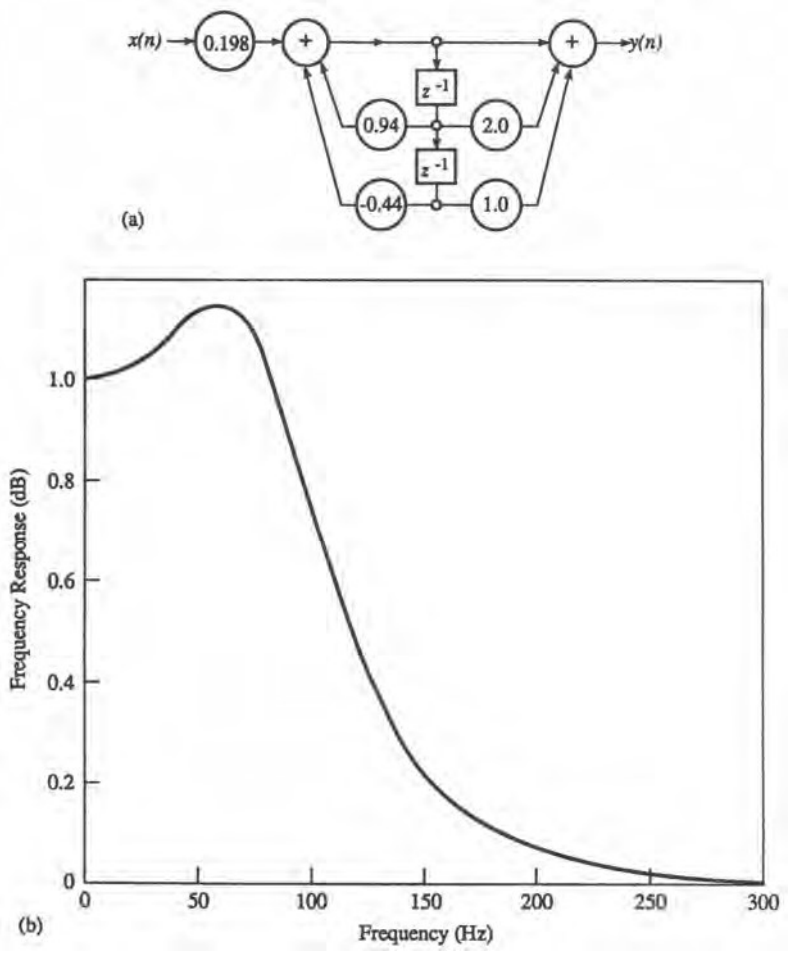


Figure 4.31. (a) Biquadratic realization of Chebyshev filter in Example 4.1.10; (b) frequency response of simulated Chebyshev filter in Example 4.1.10.

Classical filters are computationally fairly efficient since they require a minimum number of delay elements if implemented in the time domain. However, for sharp-cutoff filters as might be used in frequency multiplexing it is more practical to use the frequency-domain implementation of an ideal FIR filter for simulation purposes (see Sections 4.1.1.2 and 4.1.2.3).

4.1.4. Effects of Finite Word Length in Simulation of Digital Filters

The simulation algorithms for linear filters in IIR and FIR configurations are usually realized as programs on general-purpose computers. In both cases sequence values and coefficients are stored as finite-length floating-point numbers. This finite-word-length constraint produces quantization of the simulated values and is manifested in a variety of ways. The effect of this quantization on sampling rates of signals, structures of IIR filters, and discrete Fourier transforms will be briefly described below. Quantization effects in fixed-point arithmetic arise only in modeling of real hardware signal processors and are not covered in this section. The interested reader can find in-depth coverage of the subject in the vast literature available for digital signal processing, e.g., Refs. 1 and 2.

4.1.4.1. Roundoff Noise in Simulations of IIR Filters

The basic arithmetic operations in the simulation of an IIR filter are multiplication by a constant (the filter coefficients) and addition. It can be shown that both addition and multiplication lead to quantization noise that is proportional to the output of the operation.

The most damaging effect in the simulation of IIR filters results from the quantization of the filter coefficients. In the general approach to the analysis and synthesis of IIR filters with finite-precision coefficients, the effects of coefficient quantization can be represented by a stray transfer function in parallel with the corresponding ideal filter. The expected mean-square difference between the frequency response of the actual and ideal filter can be readily evaluated.

Knowles and Olcayto[14] formulated the coefficient quantization problem as a statistical one. They showed convincingly that for any reasonably complex filter with steep transitions between passband and stopband, the use of the *direct form* (see Section 4.1.3.1) should be avoided because of the extreme sensitivity problem. The statistical model as derived by Knowles and Olcayto predicted that the *parallel form* had the smallest degradation of performance for a fixed number of bits, followed by the *cascade connection* and then the *direct form.* The relationships between the errors and the filter parameters for the different filter configurations are quite complex and are beyond the scope of this book. Because of the results mentioned, it is recommended that for high-order filters, the biquadratic configuration should be used. It is also recommended that the coefficients should be declared as double-precision numbers and the simulation of the filter should be performed in double-precision arithmetic. The biquadratic coefficients should be computed directly from the pole and zero values. It is obvious that first obtaining the transfer function of a filter as a ratio of polynomials and subsequently factoring to obtain biquadratic forms will degrade the accuracy of the coefficients.

It should be noted that in the design of hardware digital filters, optimization techniques have been developed that optimize the performance of filters for any given truncation.

4.1.4.2. Roundoff Noise in Simulations of FIR Filters

The output of an FIR filter can be represented as a sum

$$w(n) = y(n) + f(n)$$

where $y(n)$ is the output of the ideal filter and $f(n)$ represents the effects of rounding of floating-point products and sums. It can be shown that $f(n)$ has zero mean and the signal-to-noise ratio is

$$\frac{\sigma_f^2}{\sigma_y^2} \leq (N+1)\frac{2^{-2b}}{3}$$

where σ_f^2 is the variance of $f(n)$, σ_y^2 is the variance of $y(n)$, N is the number of taps in the filter, and b is the number of bits in mantissa of the floating-point number. This shows that increasing the number of taps N in the impulse response will decrease accuracy. If this causes a problem, it is recommended that, for large N, double-precision arithmetic be used.

4.1.4.3. Effects of Quantization in Computation of the Fast Fourier Transform

The FFT is widely used in frequency-domain filtering and spectral analysis. It is important, therefore, to understand the effects of finite register length in FFT calculations[2].

If we assume that the input spectrum of the input signal is white, we can express the noise-to-signal ratio as

$$\frac{E[|F(k)|^2]}{E[|X(k)|^2]} = 2\nu\sigma_\epsilon^2$$

where $E[|F(k)|^2]$ is the expected value of the noise power caused by the mantissa roundoff, $E[|X(k)|^2]$ is the expected value of the signal power, $\nu = \log_2 N$, where N is the number of points in the FFT, and σ_ϵ^2 is the variance of the roundoff noise for each operation of multiplication and addition. Since σ_ϵ^2 is proportional to 2^{-2b}, where b is the number of bits in the mantissa, quadrupling ν (which is equivalent to raising N to the fourth power) results in an increase in noise-to-signal ratio of one bit in the mantissa. Thus, the increase of the noise-to-signal ratio as a function of N is relatively low.

4.1.5. Summary of the Process of Mapping Continuous Signals and Systems into Discrete Signals and Systems for Simulation

4.1.5.1. Introduction

It is apparent from previous sections that there are several methods available with which continuous LTI systems can be modeled for discrete simulation. The selection of the appropriate method is partially dictated not only by the system specifications and by the type of information that we expect to obtain from the simulation, but also by the computational efficiency of the method and by the personal preference of the user. A short recommendation guide to how one selects an appropriate simulation method for a given problem is described below.

To prepare a communication system and the associated signals for simulation we have to perform two main operations: transform the system and signals into lowpass equivalents, and determine the sampling rate of the simulation.

1. *Mapping into a Lowpass-Equivalent System.* Quite often the communication systems and the associated signals are of the carrier-modulated type. It is computationally inefficient to include the carrier into the simulation process. Therefore, *all* linear and most nonlinear systems are, without any loss of generality, simulated using lowpass-equivalent signals and systems. In general these modulated signals and systems have both an in-phase and quadrature component. The lowpass-equivalent signals and system will, therefore, be complex.

The mapping from carrier-modulated systems into lowpass-equivalent systems is described in general terms in Section 3.4 and for systems described by rational functions in s in Section 4.1.1.4. In most cases the carrier modulated systems and the associated signals are narrowband. The mapping in this case is straightforward. The quadrature component maps into the imaginary part and the in-phase component maps into the real part of the complex signal or system.

2. *Determining the Sampling Frequency.* The sampling frequency of a strictly bandlimited signal has to be at least twice the bandwidth of the signal (see the sampling theorem, Section 3.5). In practice, physical signals or systems are never strictly bandlimited. Usually the bandwidth is determined as the frequency band outside of which the power or energy in the signal is negligible, e.g., below –60dB. This frequency band is normally determined from the energy spectrum for aperiodic signals and from the power spectrum for periodic signals. The determination of the power spectrum for random signals is described in Chapter 6. Generally, the sampling rate is chosen to control aliasing to tolerable levels. As a rule of thumb, 8–16 samples/Hz is adequate. Higher sampling rates may be required in particular cases, for example, in feedback loops (see Chapter 8).

Many communication system simulators use a single sampling rate for the whole system. The sampling rate is thus determined by the subsystem with the largest bandwidth. When determining the largest bandwidth, special attention should be paid to subsystems with nonlinearities, as described in Chapter 5.

4.1.5.2. A Guide to the Selection of the Proper Method for Filter Simulation

To summarize this section, we describe below and in the flowchart of Figure 4.32 the process by which an appropriate method for simulation of filters is selected. The corresponding sections in this chapter that describe these methods are also indicated in the chart.

1. *Finite Impulse Response Method.* If the filter shape is specified in the frequency domain by a finite set of values in tabular form, we have two choices, either to compute the impulse response and perform time-domain convolution, or use the equivalent and more efficient method of frequency-domain filtering via the FFT. The latter method lends itself especially well to modeling filters from frequency response data. As was also seen, FFT-based frequency-domain filtering retains its efficiency even for long-duration-impulse responses (i.e., sharp-cutoff filters) as long as that duration remains a relatively small fraction of the segment length in either the OA or OS method.

Since FIR filters do not have to be physically realized before the simulation, or even to be realizable, this method is invaluable in exploring system performance for different shapes of the amplitude and phase of the channel, e.g., a Nyquist filter with zero phase.

If the filter is specified in the time domain, we perform the same operations as just mentioned, but in reverse order. We have two choices, either perform time-domain convolution or compute the frequency response and use frequency-domain filtering via the FFT. If the filter is initially IIR, then its impulse response must first be truncated.

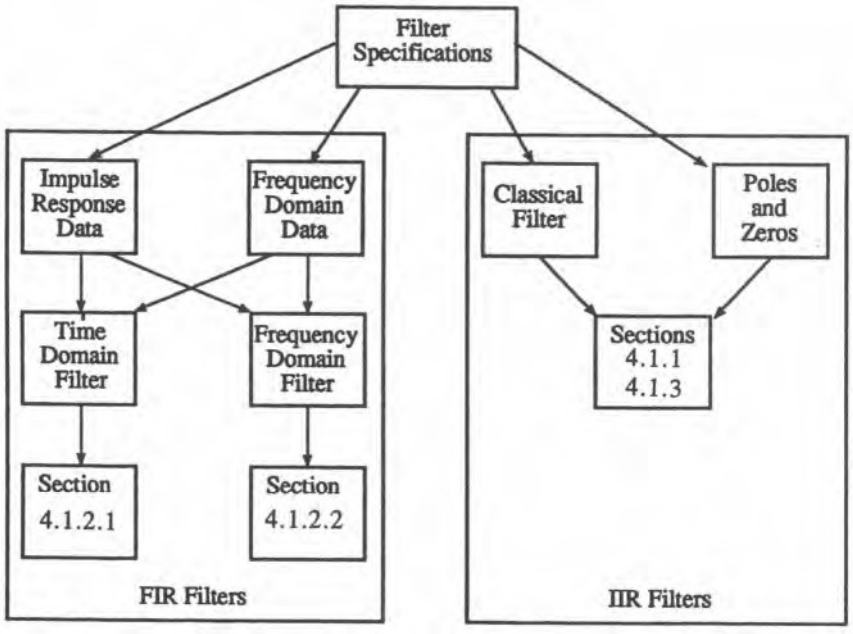


Figure 4.32. Selection of a simulation method for filters.

One advantage of FIR filters relative to IIR structures is unconditional stability. The main disadvantage of FIR filtering implemented with FFT techniques is the need to use block processing, which introduces a virtual delay in the simulation stream, thereby preventing its use in feedback loops.

2. *Infinite Impulse Response Method.* If the filter shape is specified by a classical filter, e.g., a fifth-order lowpass elliptic filter, or given as a transfer function in the form of a rational function for a designed filter, then the natural method to use is that of recursive filter simulation. Since these filters are modeled by a rational function, they are also computationally very efficient, because it takes far fewer delay elements to characterize a rational function as compared to a polynomial fit.

3. *Combination of Approaches.* The modeling choices described above are not unique and have a large area of overlap. For example, we can obtain the frequency response of a classical filter from the transfer function and use the FFT approach for simulation, either by truncating the impulse response or the frequency response, or we can fit a rational function to a specified frequency response and use a recursive filter simulation.

4.2. Time-Varying Linear Systems

Linear elements in a communication system can be time-invariant or time-varying in nature. Linear time-invariant (LTI) models and their application to the modeling and simulation of such elements as filters in communication systems were discussed in detail in the preceding section and in Chapter 3. We now turn our attention to the modeling and simulation of time-varying components.

The assumption of time invariance implies that the properties of systems being modeled do not change over (long periods of) time. If we are modeling a system by a time-invariant transfer function, it is of course implied that the transfer function remains fixed and shows no changes as a function of time. If, for example, the model is a third-order Butterworth filter with a 3-dB bandwidth of 1MHz, time invariance implies that element values and consequently the bandwidth do not change as a function of time. In order for this assumption to hold, the values of all physical components of the filter such as the resistors and active components cannot change as a function of time. While this assumption might hold over a short period of time that might be long relative to the rate of change of the information process, component values do change, albeit slightly, as a function of time due to environmental variations or aging, particularly when observed over a very long period of time. Thus, while time invariance may be a good assumption over some limited time interval, it will not hold indefinitely.

Whether to use a time-invariant or time-varying model is usually determined by the rate at which the characteristics of the communication system being modeled are changing in comparison to other parameters of the communication system such as the symbol rate. If the time constant associated with the time variations is very large compared to, say, the symbol rate in the system, then a (locally) time-invariant model may be justified. On the other hand, if the system parameters are changing at a rate approaching the symbol rate, a time-varying model is appropriate. Thus, one has *slow* or *fast* variations compared to the symbol rate or some other attribute of the system that influences the choice of a time-varying or time-invariant model. Let us further examine this point using two examples.

4.2.1. Examples of Time-Varying Systems

Consider a microwave radio communication system in which the transmitting and the receiving antennas are located on fixed microwave towers. The *channel* for these antennas is the atmosphere and the changes in the channel characteristics are due to changes in the atmospheric conditions, which typically have a time constant of several minutes to hours. Now, if the communication link is operating at a symbol rate of 100Mbit/s, then the time constant associated with the channel variations is very long compared to the symbol time. Indeed, the channel will remain in nearly the same state while billions of symbols flow over the link. Now, if the objective of simulation is BER estimation, then, during the estimation interval, the channel can be assumed to be in a *static* state and a time-invariant model can be used during the BER estimation. The long-term behavior of the channel and its impact on long-term system performance can be evaluated by analyzing system performance over a series of *snapshots* of static channel conditions, using a different time-invariant model for each snapshot. From such simulations, one can obtain performance measures such as *outage probabilities* which describe the portion of time during which the channel performance might fall below some BER threshold.

As a second example, consider a mobile communication system which consists of a fixed base station and a mobile user. The characteristics of the communication channel between the transmitter and the receiver in this system will be changing as a function of time since the parameters of the channel such as the attenuation and delay are changing because of relative motion between the base station and the user, in addition to changes in atmospheric conditions. If the mobile user is moving fast and if the symbol rate is of the order of 10 Ksymbols/s, then the rate at which channel conditions change might be comparable to the symbol rate and hence we need to consider a time-varying channel model. While a time-

varying model may not be needed for BER estimation, such a model will be necessary to study the behavior of receiver subsystems such as synchronizers and equalizers.

4.2.2. Time-Domain Description for Linear Time-Varying Systems

4.2.2.1. The Impulse Response

The commonly used form of the time-varying impulse response is modeled on the definition of the impulse response for linear time-invariant systems defined in Chapter 3,

$$h(t, \tau) = \Gamma[\delta(t - \tau)] \tag{4.2.1a}$$

where τ is the time (from the origin) when the impulse is applied, and Γ is the linear time-varying system operator.

The most important application of time-varying models in communication systems is in the modeling of time-varying (multipath) channels. To describe the behavior of such channels, Kailath[15] introduced several alternative formulations for the impulse response, one of which is analytically more convenient than the others in different situations. The most convenient for purposes of modeling communication channels is to define $c(\hat{\tau}, t)$ as the response measured at time t to a unit impulse applied at time $t - \hat{\tau}$,

$$c(\hat{\tau}, t) = \Gamma[\delta(t - (t - \hat{\tau}))] = \Gamma[\delta(\hat{\tau})] \tag{4.2.1b}$$

We will refer to the above impulse response as the *channel impulse response.*

Note that the relationship of the above definition of the impulse response to the channel impulse response is $c(\hat{\tau}, t) = h(t, t - \tau)$ or $h(t, \tau) = c(t - \hat{\tau}, t)$. We will use one or the other definition of the above impulse responses in this section, depending on which is most appropriate to use in a given situation.

Since the system is time-varying, the impulse response will change as a function of both the time at which the impulse is applied and the time at which the output is measured. Hence, both $c(\hat{\tau}, t)$ and $h(t, \tau)$ are surfaces in three-dimensional space. Cross sections of these surfaces are illustrated in Figure 4.33. Figure 4.33a shows two hypothetical impulse responses $h(t, \tau)$ for values of $\tau = 0, 2$. Observe that causality requires $h(t, \tau) = 0$ for $t < \tau$. Because of the relationships $c(\hat{\tau}, t) = h(t, t - \tau)$ and $\hat{\tau} = t - \tau$, it is simple to establish a correspondence between the two functions. For example, the cross section $h(t, 2)$ can be mapped to a cross section $c(\hat{\tau}, t)$ with the help of the following table:

t	τ	$\hat{\tau}$
2	2	0
3	2	1
4	2	2
5	2	3
.	.	.
.	.	.
.	.	.

Thus, for example, $h(5,2)$ is numerically equal to $c(3,5)$. Hence, the cross sections of Figure 4.33a correspond to the cross sections in Figure 4.33b of the surface $c(\cdot,\cdot)$. In order to sharpen the distinction between h and c, we have used the symbols τ and $\hat{\tau}$ ($\hat{\tau} = t - \tau$). In Chapter 9,

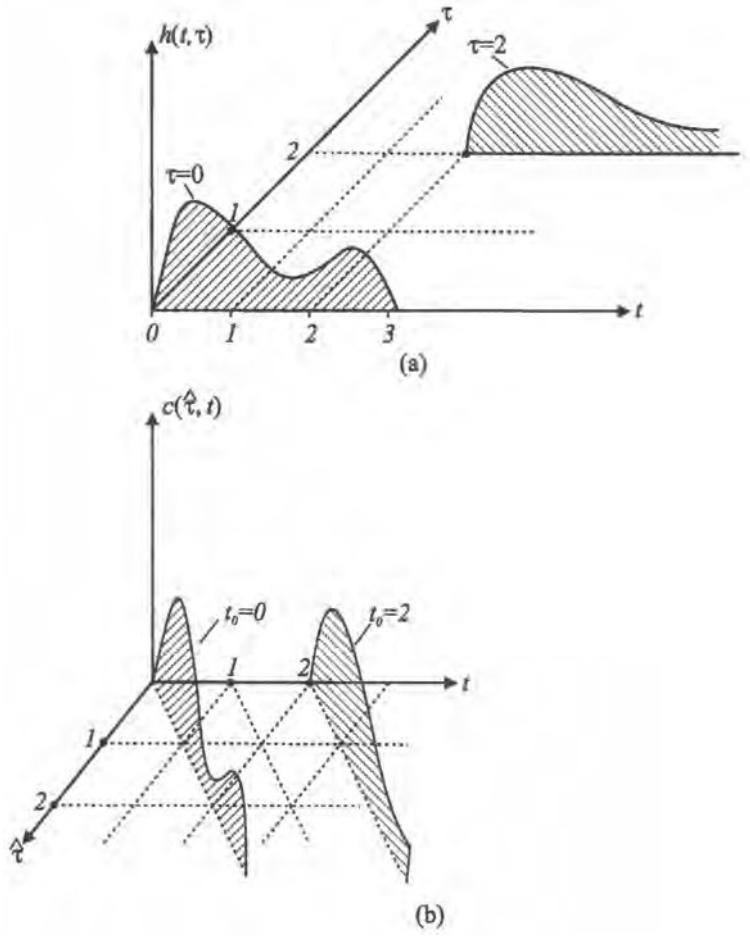


Figure 4.33. Two-dimensional representation of a time-varying impulse response. **(a)** $h(t, \tau)$; **(b)** $c(\hat{\tau}, t)$.

where we use c extensively, we will use the simpler notation τ for what we labeled here $\hat{\tau}$. For a time-invariant system the impulse response will strictly be a function of the elapsed time and

$$h(t_1, \tau) = h(t_2, \tau) = \cdots = h(t - \tau) \text{ or } c(\hat{\tau}, t_1) = c(\hat{\tau}, t_2) = \cdots = c(\hat{\tau})$$

While the impulse response of an LTI system maintains the same functional form irrespective of when the impulse was applied at the input, the impulse response of an LTV system depends on when the input was applied, as mentioned above. An illustration of this point is given in Figure 9.4 in Chapter 9.

4.2.2.2. The Superposition Integral

The response of the system to an arbitrary input is determined by a superposition integral[15,16,17] (actually, a Fredholm integral[18]). Using the impulse response, the superposition integral is

$$y(t) = \int_{-\infty}^{\infty} h(t, \tau)x(\tau)\, d\tau \qquad \text{or} \qquad y(t) = h(t, \tau) * x(\tau) \tag{4.2.2a}$$

In terms of the channel response we have

$$y(t) = \int_{-\infty}^{\infty} c(\hat{\tau}, t)x(t - \hat{\tau})\, d\hat{\tau} \qquad \text{or} \qquad y(t) = c(\hat{\tau}, t) * x(t - \hat{\tau}) \tag{4.2.2b}$$

When the system is time-invariant, $h(t, \tau) = h(t - \tau)$ and $c(\hat{\tau}, t) = c(\hat{\tau})$, and (4.2.2) is the convolution integral. Although the convolution integral is also sometimes referred to as the superposition integral, we will reserve the term *superposition* for the time-varying case in (4.2.2). For causal systems this integral may be written as

$$y(t) = \int_{0}^{\infty} h(t, \tau)x(\tau)\, d\tau \tag{4.2.3a}$$

because $h(t, \tau) = 0$ for $\tau < 0$, and

$$y(t) = \int_{0}^{\infty} c(\hat{\tau}, t)x(t - \hat{\tau})\, d\hat{\tau} \tag{4.2.3b}$$

where $c(\hat{\tau}, t) = 0$ for $\hat{\tau} > t$.

4.2.3. Frequency-Domain Representations of Time-Varying Systems

For an LTV system one can develop the notion of a transfer function,[19] albeit a time-varying transfer function $C_{\hat{\tau}}(f, t)$, by simply taking the Fourier transform of $c(\hat{\tau}, t)$ with respect to $\hat{\tau}$ as

$$C_{\hat{\tau}}(f, t) = \int_{-\infty}^{\infty} c(\hat{\tau}, t)e^{-j2\pi f\hat{\tau}} d\hat{\tau} \tag{4.2.4}$$

If the system is slowly time-varying, then the concepts of frequency response and bandwidth can be applied to $C_{\hat{\tau}}(f, t)$. Whereas the LTI system is characterized by a single impulse response function and a single transfer function, the LTV system is characterized by a family of impulse response functions and transfer functions, one function for each value of t, as shown in Figure 4.33. In fact, $C_{\hat{\tau}}(f, t)e^{2\pi ft}$ is the response to $\exp([j2\pi ft)$.

The variable t in $c(\hat{\tau}, t)$ and $C_{\hat{\tau}}(f, t)$ describes the time-varying nature of the system. Strong dependence on t and fast changes associated with t indicate a rapidly time-varying system. Usually, the time-varying nature of the system is modeled as a random phenomenon and is treated as a random process in t (see multipath channels in Section 9.1.3). If the process is stationary, then the time variations can be modeled by an appropriate autocorrelation function in the time domain or by a corresponding power spectral density in the frequency domain. The time constant of the autocorrelation function or the bandwidth of the power spectral density are key parameters that describe whether $c(\hat{\tau}, t)$ is slowly or rapidly time-varying.

The inverse transform is given by

$$c(\hat{\tau}, t) = \int_{-\infty}^{\infty} C_{\hat{\tau}}(f, t)e^{2\pi f\hat{\tau}}\, df \tag{4.2.5}$$

The output $y(t)$ of a system $c(\hat{\tau}, t)$ with the input $x(t)$ can be determined by

$$y(t) = \int_{-\infty}^{\infty} C_{\hat{\tau}}(f, t)X(f)e^{2\pi ft}\, df \tag{4.2.6}$$

where $X(f)$ is the Fourier transform of $x(t)$. The time-varying frequency response describes the system completely, as does the impulse response $c(\hat{\tau}, t)$.

4.2.3.1. *Two-Dimensional Frequency Response*

Kailath[15] introduced a frequency-domain function

$$C_{\hat{\tau},t}(f, \nu) = \int_{-\infty}^{\infty} C_{\hat{\tau}}(f, t)e^{-j2\pi\nu t}dt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} c(\hat{\tau}, t)e^{-j2\pi f\hat{\tau}}e^{-j2\pi\nu t}\, d\hat{\tau}\, dt \qquad (4.2.7)$$

The frequency v is associated with the rate of change of the system and f is associated with the frequency of the system excitation.

In defining the above two-dimensional Fourier transform, the usual finite-energy assumption is made about the impulse response function. However, we will see later in Section 9.4 that for channel modeling, $c(\hat{\tau}, t)$ will be treated as a stationary random process in t, in which case the Fourier transform of $c(\hat{\tau}, t)$ may not exist with respect to t. The appropriate procedure here is to define the autocorrelation of $c(\hat{\tau}, t)$ with respect to the t variable and then take the Fourier transform of the autocorrelation function to obtain the frequency-domain representation in terms of the power spectral density of the random process.

From (4.2.5) and (4.2.6) we get the system output frequency response

$$Y(\nu) = \int_{-\infty}^{\infty} C_{\hat{\tau},t}(f, \nu - f)X(f)\, df \qquad (4.2.8)$$

Equation (4.2.8) is the frequency-domain convolution of the input and system frequency-domain characteristics. In the double integral of Equation (4.2.7), the frequency variable f is associated with the tune variable $\hat{\tau}$ and it may be viewed as analogous to the frequency variable f in the transfer function $H(f)$ of linear time-invariant systems. However, the input-output relationship for LTV systems in the frequency domain given in Equation (4.2.8) involves a convolution in the frequency domain in the second variable of the transfer function $C(f, v)$. This convolution accounts for the effect of the time-varying nature of the system in the frequency domain.

4.2.3.2. *Bandwidth Relations in Time-Varying Systems*

The time-varying character of systems is usually manifested by a frequency spread, or a frequency shift, or both. Thus, the response of a LTV system to a single frequency f_0 can be a frequency spectrum or a frequency spectrum centered about a different frequency than f_0 (see Figure 4.34). The width of the spectrum (or frequency spread) about f_0, appropriately defined, can be regarded as a measure of the rate of variation of the system. This frequency spread will depend on f_0. An input signal normally has some nonzero bandwidth. Hence the frequency spread may differ for every spectral component. However, we can reasonably define the largest spread B_s as an appropriate measure of the time variability of the system. This spreading is usually encountered in multipath channel models, where it is usually referred to as Doppler spread, and the shift of the center frequency is referred as the Doppler shift.

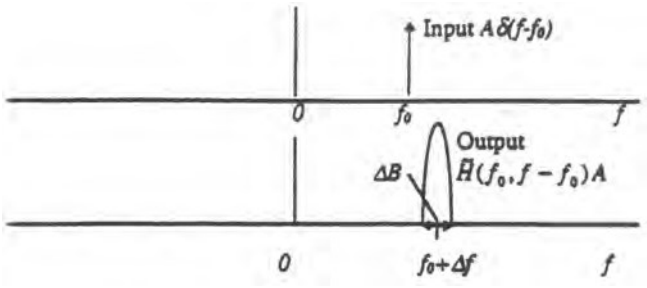


Figure 4.34. Illustration of frequency shift and spectral spreading due to motion of the transmitter, receiver, or medium.

4.2.3.3. Sampling Rate

From (4.2.8) we notice that an input with bandwidth B_i to a linear time-varying system characterized by B_s results in an output bandwidth not greater than $B_i + B_s$. Hence, to eliminate aliasing distortion in simulations of time-varying linear systems, the sampling rate f_s has to satisfy the sampling theorem (see Section 3.5), so that

$$f_s = \frac{1}{T_s} \geq 2(B_i + B_s) \tag{4.2.9}$$

In many cases $B_i >> B_s$ and there is little impact on f_s, but when that is not the case, as in very low data rate systems, the simulation user should make sure that f_s is large enough to take the spread into account.

4.2.4. Properties of Linear Time-Varying Systems

4.2.4.1. Introduction

The properties of convolution for LTI systems were described in Section 2.2. Similar to the time-invariant case, the superposition possesses the properties of *associativity*

$$h_1(t,\tau) * [h_2(t,\tau) * h_3(t,\tau)] = [h_1(t,\tau) * h_2(t,\tau)] * h_3(t,\tau) \tag{4.2.10a}$$

and *distributivity*

$$h_1(t,\tau) * [h_2(t,\tau) + h_3(t,\tau)] = h_1(t,\tau) * h_2(t,\tau) + h_1(t,\tau) * h_3(t,\tau) \tag{4.2.10b}$$

However, generally it does not possess the *commutative* property

$$h_1(t,\tau) * h_2(t,\tau) \neq h_2(t,\tau) * h_1(t,\tau) \tag{4.2.10c}$$

4.2.4.2. Interconnections of Linear Time-Varying Systems

As in the case of LTI systems described in Section 3.6.5, the interconnections of LTV systems can be simplified using block diagram equivalents that are similar to the ones shown in Figure 3.25. However, the simplification of block diagrams for LTV systems is much more complicated than for LTI systems since transform methods are not applicable. Therefore, for the time-varying case the operations indicated in Figure 3.25 are not simply additions and multiplications of transfer functions as in the time-invariant case. Rather, the block-diagram

equivalents apply formally to the time-varying case by replacing H_1 by Γ_1, H_2 by Γ_2, and H by Γ, where the Γ's are time-dependent linear operators whose properties may not always allow the usual simplification of block diagrams.

Cascade Interconnection of Systems. As an example of the complexity of reducing block diagrams of a time-varying system, consider a cascade connection of LTV systems Γ_1 and Γ_2. Let the impulse responses of systems Γ_1 and Γ_2 be $h_1(\tau, t)$ and $h_2(\tau, t)$, respectively.

Then

$$\begin{aligned} z(t) &= \int_{-\infty}^{t} h_1(t, \tau)x(\tau)\, d\tau \\ y(t) &= \int_{-\infty}^{t} h_2(t, \xi)z(\xi)\, d\xi \end{aligned} \tag{4.2.11}$$

Combining these equations, we get

$$\begin{aligned} y(t) &= \int_{-\infty}^{t} d\xi\, h_2(t, \xi) \int_{-\infty}^{\xi} h_1(\xi, \tau)x(\tau)\, d\tau \\ &= \int_{-\infty}^{t} d\tau\, x(\tau) \int_{\tau}^{t} h_2(t, \xi)h_1(\xi, \tau)\, d\xi \end{aligned} \tag{4.2.12}$$

Therefore the equivalent system $\Gamma = \Gamma_1\Gamma_2$ has the impulse response

$$h_1 * h_2 = h(t, \tau) = \int_{\tau}^{t} h_2(t, \xi)h_1(\xi, \tau)\, d\xi \tag{4.2.13}$$

A cascade interconnection of systems is equivalent to the superposition of their impulse responses. It should be noted, however, that the cascade operation is not commutative. Indeed, if we reverse the order of the cascade connection, the equivalent system is given by

$$h_2 * h_1 = \int_{\tau}^{t} h_1(t, \xi)h_2(\xi, \tau)\, d\xi \tag{4.2.14}$$

■ *Example 4.2.1.* The system in Figure 4.35 has

$$h_1(t, \tau) = u(t - \tau); \qquad h_2(t, \tau) = t\delta(t - \tau)$$

The superposition for the cascade connection, using (4.2.13), is given by

$$h_1 * h_2 = \int_{\tau}^{t} t\delta(t - \xi)u(\xi - \tau)\, d\xi = tu(t - \tau)$$

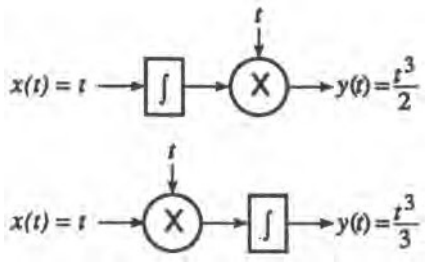


Figure 4.35. Illustration of the noncommutativity of time-varying linear systems (for Example 4.2.1).

while the reverse cascade connection yields

$$h_2 * h_1 = \int_{\tau}^{t} u(t-\xi)\xi\delta(\xi-\tau)\,d\xi = \tau u(t-\tau)$$

Applying the input $x(t) = t$ for $t \geq 0$ and using (4.2.2a) yields $y(t) = t^3/2$ in the first instance and $t^3/3$ in the second. ■

Other interconnection simplifications that involve a system function inverse, as in Figure 3.25c and especially the elimination of feedback interconnections in Figure 3.25e, would greatly simplify the simulation of communication systems. However, the evaluation of the inverse of time-varying systems is usually very difficult and its use in simulations is therefore limited.[20]

4.2.5. Models for LTV Systems

4.2.5.1. Linear Differential Equation with Time-Varying Coefficients

Some LTV systems are described by linear differential equations with time-varying coefficients (LTV/DE) of the form

$$a_n(t)\frac{d^n y(t)}{dt^n} + a_{n-1}(t)\frac{d^{n-1}y(t)}{dt^{n-1}} + \cdots + a_0(t)y(t) = x(t) \qquad (4.2.15a)$$

If the system is slowly varying, i.e., the coefficients of the LTV/DE are varying much more slowly than the signal rate, the system can be regarded as *quasistatic*. The LTV/DE can then be modeled as a recursive IIR filter described in Section 4.1.

For rapidly varying systems we now find the impulse response of the system, which is defined to be the solution of (4.2.15a) when $x(t) = \delta(t-\tau)$. It is well known[17,18] that the solution of (4.2.15a) when $x(t) = \delta(t-\tau)$ is $u(t-\tau)$ times the solution $y(t,\tau)$ of the initial value problem

$$a_n(t)\frac{d^n y(t)}{dt^n} + a_{n-1}(t)\frac{d^{n-1}y(t)}{dt^{n-1}} + \cdots + a_0(t)y(t) = 0 \qquad (4.2.15b)$$

with the initial conditions

$$y(\tau) = y'(\tau) = \cdots = y^{(n-2)}(\tau) = y^{(n)}(\tau) = 0, \qquad y^{(n-1)}(\tau) = \frac{1}{a_n(\tau)}$$

Therefore the impulse response is

$$h(t,\tau) = y(t,\tau)u(t-\tau) \qquad (4.2.16)$$

If a LTV/DE has an analytical solution, its impulse response has a separable form

$$h(t,\tau) = \sum_{i=1}^{n} p_i(t)q_i(\tau), \qquad t \geq \tau \qquad (4.2.17)$$

The above represents one of the most desirable types of models (known as a *separation model*) to be used both in simulation and analysis of time-varying linear systems.

Unfortunately, only first-order LTV/DEs have a general solution. Therefore, most "rapidly varying" LTV/DEs are modeled and simulated using the same methods that are used for nonlinear differential equations in Section 5.1.

As an illustration of an analytical evaluation of an LTV/DE, we evaluate the impulse response for a first-order equation

$$a_1(t)\frac{dy(t)}{dt} + a_0(t)y(t) = \delta(t-\tau) \tag{4.2.18}$$

The first step is to find a solution to the homogeneous equation

$$a_1(t)\frac{dy(t)}{dt} + a_0(t)y(t) = 0$$

Rearranging terms, we get

$$\frac{dy(t)}{y(t)} = -\frac{a_0(t)}{a_1(t)}dt$$

and therefore

$$y(t,\tau) = C\exp\left[-\int_0^t \frac{a_0(\xi)}{a_1(\xi)}d\xi\right]$$

Imposing the initial condition $y(\tau,\tau) = 1/a_1(\tau)$ allows us to determine C, and we find

$$h(t,\tau) = \frac{1}{a_1(\tau)}\exp\left[\int_0^\tau \frac{a_0(\xi)}{a_1(\xi)}d\xi - \int_0^t \frac{a_0(\xi)}{a_1(\xi)}d\xi\right]u(t-\tau) \tag{4.2.19}$$

An example of a first-order impulse response is shown in the following example.

■ *Example 4.2.2.* Determine the impulse response of the LTV system defined by

$$(t+1)\frac{dy(t)}{dt} + (t^2-1)y(t) = \delta(t-\tau)$$

From (4.2.19)

$$h(t,\tau) = \frac{1}{\tau+1}\exp\left[\int_0^\tau \frac{\xi^2-1}{\xi+1}d\xi - \int_0^t \frac{\xi^2-1}{\xi+1}d\xi\right]u(t-\tau)$$

which yields

$$h(t,\tau) = \frac{1}{\tau+1}\exp\left[\frac{\tau^2}{2}-\tau\right]\exp\left[-\frac{t^2}{2}+t\right]u(t-\tau)$$

The impulse response $h(t,\tau)$ for four values of τ is shown in Figure 4.36. ■

4.2.5.2. *Separable Models*

As mentioned before, among the most desirable types of models to be used both in simulation and analysis of time-varying linear systems are the ones that have separable impulse responses.[21] † An impulse response is termed separable if

$$h(t,\tau) = \sum_{i=1}^{n} p_i(t)q_i(\tau), \qquad t \geq \tau$$

† It can be shown[22] that all impulse responses of linear systems characterized by a scalar linear time-varying differential equation as in (4.2.15a) are separable. However, as noted earlier, solutions to differential equations other than first order are generally not available.

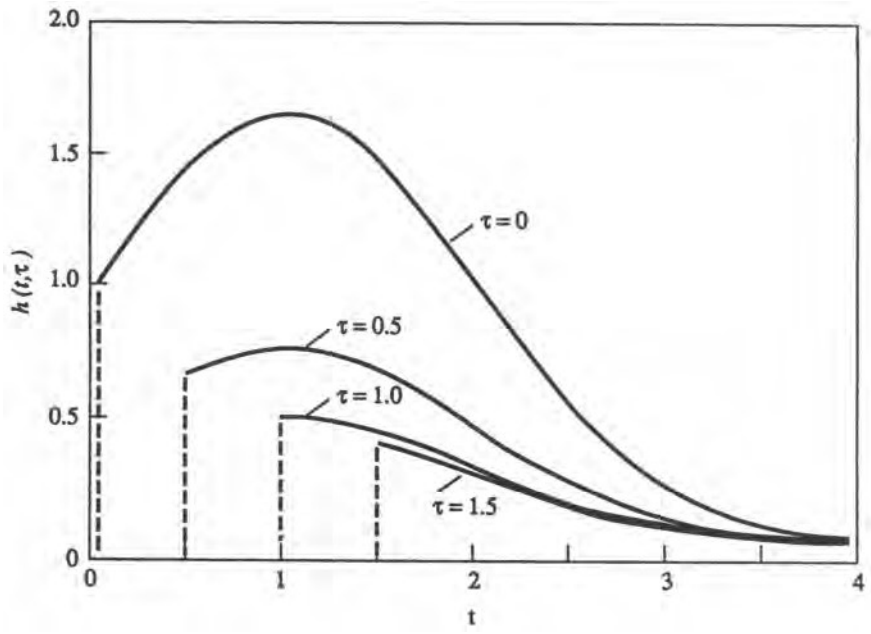


Figure 4.36. Equivalent linear time-varying system (for Example 4.2.2).

The system output for a separable impulse response is thus

$$y(t) = \int_0^t h(t, \tau)x(\tau)\, d\tau = \sum_{i=1}^{N} p_i(t) \int_0^t x(\tau)q_i(\tau)\, d\tau \qquad (4.2.20)$$

The realization of such a system is shown in Figure 4.37.

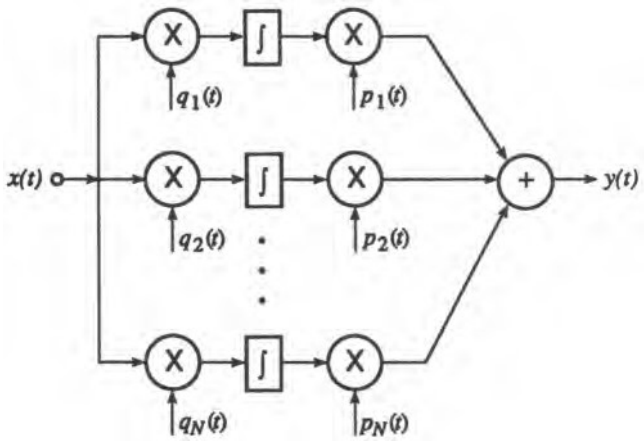


Figure 4.37. Structure of separable models for linear time-varying systems.

4.2.5.3. Tapped Delay-Line Channel Models

A variety of so-called *tapped delay-line* (TDL) models have been developed by Kailath based on the sampling theorem[15]. These models differ depending upon whether one assumes the input to the channel to be bandlimited or the output to be bandlimited, and in addition, whether or not one assumes the channel itself to be bandlimited. For our purposes, probably the most useful of the models is the one derived from assuming that the input signal is bandlimited.

From this formulation, a tapped delay-line model can be derived either for lowpass or bandpass channels. We confine ourselves here to a lowpass channel since it can be applied directly to the lowpass-equivalent case. The transmitted signal is modeled as

$$x(t) = s(t) * h(t) \tag{4.2.21}$$

where * represents convolution, $s(t)$ is the modulator output, and $h(t)$ is the impulse response of an ideal lowpass output filter with transfer function $H(f) = 1$ for $|f| \leq B_i$ and $H(f) = 0$ for $|f| > B_i$. Thus, the channel input is bandlimited to B_i; of course, if we assume $s(t)$ itself to be bandlimited with bandwidth $\leq B_i$, then the lowpass filter is superfluous.

There are a number of ways to arrive at the desired result. It is interesting to consider two different approaches. One of them will be given in Chapter 9, in conjunction with *randomly* time-varying channels. Here, we take a slightly different path, as follows.

First, based on the bandlimited assumption on $x(t)$, we can write it in the form of the sampling expansion. As we discussed earlier, there are two variables, $\hat{\tau}$ and t, the first associated with signal variations, and the second with the system or channel variations. It will be convenient, therefore, to use the $\hat{\tau}$ variable to expand $x(\cdot)$. Thus, from (3.5.9),

$$x(\hat{\tau}) = \sum_{n=-\infty}^{\infty} x\left[\frac{n}{2B_i}\right] \operatorname{sinc}(2B_i\hat{\tau} - n) \tag{4.2.22}$$

Now, the Fourier transform of $x(\hat{\tau})$ is bandlimited by definition:

$$X(f) = F[x(\hat{\tau})] = 0 \qquad \text{for} \quad |f| > B_i$$

Hence, the system output (in the frequency domain)

$$Y(f, t) = C(f, t)X(f) \tag{4.2.23}$$

is also bandlimited to $|f| \leq B_i$. Therefore, (4.2.23) is equivalent to

$$Y(f, t) = G(f, t)X(f) \tag{4.2.24}$$

where $G(f, t)$ is the channel's effective transfer function with respect to the signal:

$$G(f, t) = C(f, t)H(f) \tag{4.2.25}$$

where $H(f)$ is an ideal filter with bandwidth B_i. Thus, $G(f, t)$ is also bandlimited, and its transform $g(\hat{\tau}, t)$ is also representable in terms of the sampling expansion:

$$g(\hat{\tau}, t) = \sum_{m=-\infty}^{\infty} g\left[\frac{m}{2B_i}, t\right] \operatorname{sinc}(2B_i\hat{\tau} - m) \tag{4.2.26}$$

Applying (4.2.2b), we find that (4.2.24) implies

$$y(t) = \int_{-\infty}^{\infty} g(\hat{\tau}, t)x(t - \hat{\tau})\, d\hat{\tau} \tag{4.2.27}$$

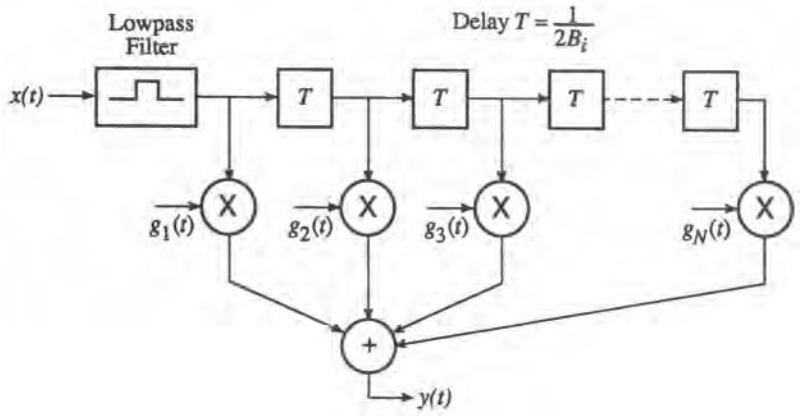


Figure 4.38. Sampling model for linear time-varying systems, in the form of a tapped delay-line with time-varying coefficients.

and upon substituting (4.2.22) and (4.2.26), it is straightforward to show that (4.2.27) is equivalent to

$$y(t) = \frac{1}{2B_i} \sum_{m=-\infty}^{\infty} g\left[\frac{m}{2B_i}, t\right] x\left[t - \frac{m}{2B_i}\right] \tag{4.2.28}$$

The representation (4.2.28) can be synthesized by a tapped delay-line with taps having time-varying gains $g_n(t) = g(n/(2B_i), t)$ and tap delay $T = (2B_i)^{-1}$, as shown in Figure 4.38.

Theoretically, the delay line should have an infinite number of taps. In practice, we use a finite length delay line at the cost of an error that can be made arbitrarily small. These basic notions underline the tapped-delay-line random multipath channel models discussed in Chapter 9. In the random case, $c(\hat{\tau}, t)$ is a random process in t and must be given a statistical characterization; the tap gains $g_n(t)$ may then be correlated.

As mentioned, the tapped delay-line model (4.2.22) forms the basis for modeling multipath radio channels. For simulation application we note that a quasistatic approach is feasible if the channel Doppler spread (fading rate) is much smaller than the signal bandwidth, i.e., $B_s \ll B_i$. In this case, the tap gains can be regarded as constant for some period of time that is nevertheless long with respect to a signaling interval (or reciprocal of the bandwidth). The simulation of the system under fading conditions can then be approximated by a series of simulations in each of which the tap gains are assumed to be constant. (See Case Study I in Chapter 12.)

4.3. Summary

In this chapter we have discussed how to go about simulating the effect of a linear system on an input signal. The synonym almost universally used for linear system is *filter*, and likewise the effect in question is commonly called *filtering*. In most systems, filters are linear and time-invariant (LTI), but they can also be linear and time-varying (LTV), as in adaptive structures. While not system elements as such, certain wireless channels are also properly modeled as (randomly) time-varying. LTI systems are much easier to deal with than LTV systems, and by far the preponderance of analytical and simulation techniques that have been developed apply to the former.

Although the modeling of a filter as such and of the filtering process will evidently be closely connected, it is useful for a clearer view of the subject to separate the two initially. Thus, we began the chapter with an outline of the ways in which LTI filters can be described or specified in practical applications and how continuous filter structures can be approximated in discrete time. Along the way, we presented the particulars for an important practical class of filters, the "classical" filters, which include the Butterworth, Chebyshev, and elliptic filters.

For modeling purposes, two major approaches present themselves, one where a filter is represented (or approximated) by a finite impulse response (FIR), and the other where the impulse response is infinite (IIR). For FIR filters, two filtering techniques are possible, referred to as time-domain and frequency-domain, respectively. Time-domain filtering is equivalent to a sliding (finite) convolution sum, and frequency-domain filtering is simply another term for using the discrete Fourier transform, normally implemented with the computationally efficient fast Fourier transform (FFT). Although it is possible to design FIR filters in the form of digital signal processors, in which case there can be no modeling error, FIR filter *models* often arise from truncating the impulse response of the actual filter. In this case errors arise both from truncation and aliasing. Both errors can be controlled (at some cost), the first by judicious use of windows, and the second by sufficiently frequent sampling.

If we wish to retain the "infinite" character of a filter's impulse response, a different set of techniques must be used both for the filter representation and the filtering itself. Here, two basic approaches are common. In the first, it is desired to duplicate the actual impulse response, subject only to the limitation of discrete sampling. This approach is called the *impulse-invariant transformation.* It is obvious, however, that even if we have a way of calculating the impulse response samples, the infinite memory of the filter cannot be literally reproduced. If the simulation starts at $t = 0$, however, we can in principle keep going indefinitely. This can be visualized as an ever-increasing convolution sum, but this will quickly become computationally burdensome, and perhaps also problematic with respect to precision. These difficulties can be mitigated, though not completely, by expressing the filter output recursively, if this is possible. For filters whose transfer functions are rational functions of s or z, this is indeed possible, as was shown.

The second approach to simulating IIR filtering is to first model the filter through what we called a transform-domain approximation. This approach is Intended to apply to filters whose transfer function is defined as a ratio of polynomials in s, or equivalently a ratio of polynomials in $1/s$. The nature of the approximation is to visualize $1/s$ as integration and replace it by a discrete integration formula. The most widely used such formula is called the *bilinear transformation*, which amounts to the trapezoidal integration formula. For such filters as we are considering, replacing $1/s$ by a ratio of polynomials in z (the bilinear transformation being a specific instance of this) leads to the implementation of the filtering process as a linear (recursive) difference equation. Standard alternative forms for realizing such filter structures (both for hardware and computation) were presented.

General hints were given for approaching the simulation of filtering both for FIR and IIR representations.

As indicated above, LTV systems are more difficult to deal with, and the options for simulating them much more limited. We provided an introductory discussion of the properties of LTV systems. From the simulation standpoint, one of the potentially more significant properties is the bandwidth expansion induced by LTV behavior, captured by the term *Doppler broadening.* Depending on the degree of bandwidth expansion relative to the data rate, this will require an increase in the simulation sampling rate. Under the reasonable assumption that the input signal is bandlimited, we also showed that an LTV system can be represented as a tapped delay-line with time-varying coefficients. This structural model is of

the most useful one for simulating LTV systems, and in Chapter 9 we shall apply it to the mobile (multipath) channel, for which case the tap coefficients become *randomly* time-varying.

4.4. Appendix: Biquadratic Factors for Classical Filters

This appendix lists biquadratic factors for normalized lowpass Butterworth filters (Table 4.A.1), normalized lowpass Bessel filters (Table 4.A.2), normalized lowpass Chebyshev filters (Table 4.A.3), and normalized lowpass elliptic filters (Table 4.A.4).

Table 4.A.1. Biquadratic Factors for Normalized Lowpass Butterworth Filters

$$H(s) = \frac{H(0)}{D(s)};\ \omega_b = 1$$

n	Denominator $D(s)$
1	$s+1$
2	$s^2+\sqrt{2}s+1$
3	$(s+1)(s^2+s+1)$
4	$(s^2+0.765s+1)(s^2+1.848s+1)$
5	$(s+1.0000)[(s+0.3090)^2+0.9511^2][(s+0.8090)^2+0.5878^2]$
6	$[(s+0.2588)^2+0.9659^2][(s+0.7071)^2+0.7071^2][(s+0.9659)^2-0.2588^2]$
7	$(s+1.0000)[(s+0.2225)^2+0.9749^2][s+0.6235)^2+0.7818^2][(s+0.9010)^2+0.4339^2]$
8	$[(s+0.1951)^2+0.9808^2][(s+0.5556)^2+0.8315^2][(s+0.8315)^2-0.5556^2]$ $\times[(s+0.9808)^2+0.1951^2]$
9	$(s+1.000)[(s+0.1737)^2+0.9848^2][(s+0.5000)^2+0.8660^2][(s-0.7660)^2+0.6428^2]$ $\times[(s+0.9397)^2+0.3420^2]$
10	$[(s+0.1564)^2+0.9877^2][(s+0.4540)^2+0.8910^2][(s+0.7071)^2-0.7071^2]$ $\times[(s+0.8910)^2+0.4540^2][(s+0.9877)^2+0.1564^2]$

Table 4.A.2. Biquadratic Factors for Normalized Lowpass Bessel Filters

$$H(s) = \frac{d_0}{D(s)};\ \tau_0 = 1$$

n	Denominator $D(s)$
1	$s+1$
2	s^2+3s+3
3	$(s+2.322)(s^2+3.678s+6.460)$
4	$(s^2+5.792s+9.140)(s^2+4.208s+11.488)$
5	$(s+3.6467)[(s+3.3520)^2+1.7427^2][(s+2.3247)^2+3.5710^2]$
6	$[(s+4.2484)^2+0.8675^2][(s+3.7356)^2+2.6263^2][(s+2.5159)^2+4.4927^2]$
7	$(s+4.9718)[(s+4.7583)^2+1.7393^2][(s+4.0701)^2+3.5172^2][(s+2.6857)^2+5.4207^2]$
8	$[(s+5.5879)^2+0.8676^2][(s+2.8390)^2+6.3539^2][(s+4.3683)^2+4.1444^2]$ $\times[(s+5.2048)^2+2.6162^2]$
9	$(s+6.2970)[(s+6.1294)^2+1.7378^2][(s+5.6044)^2+3.4982^2][(s+4.6384)^2+5.3173^2]$ $\times[(s+2.9793)^2+7.2915^2]$
10	$[(s+6.9220)^2+0.8677^2][(s+3.1089)^2+8.2327^2][(s+6.6153)^2+2.6116^2]$ $\times[(s+5.9675)^2+4.3850^2][(s+4.8862)^2+6.2250^2]$

Table 4.A.3. Biquadratic Factors for Normalized Lowpass Chebyshev Filters

$$H(s) = H_0/D(s);\ \omega_p = 1$$

n	Denominator $D(s)$
	$\alpha_p = 0.5$ dB ($\epsilon = 0.3493$)
1	$s + 2.863$
2	$s^2 + 1.425s + 1.516$
3	$(s + 0.626)(s^2 + 0.626s + 1.142)$
4	$(s^2 + 0.351s + 1.064)(s^2 + 0.845s + 0.356)$
5	$(s + 0.3623)[(s + 0.1120)^2 + 1.0116^2][(s + 0.2931)^2 + 0.6252^2]$
6	$[(s + 0.0777)^2 + 1.0085^2][(s + 0.2121)^2 + 0.7382^2][(s + 0.2898)^2 - 0.2702^2]$
7	$(s + 0.2562)[(s + 0.0570)^2 + 1.0064^2][(s + 0.1597)^2 + 0.8001^2][(s - 0.2308)^2 + 0.4479^2]$
8	$[(s + 0.436)^2 + 1.0050^2][(s + 0.1242)^2 + 0.8520^2][(s + 0.1859)^2 + 0.5693^2]$ $\times[(s + 0.2193)^2 + 0.1999^2]$
9	$(s + 0.1984)[(s + 0.0345)^2 + 1.0040^2][(s + 0992)^2 + 0.8829^2][(s - 0.1520)^2 + 0.6553^2]$ $\times[(s + 0.1864)^2 + 0.3487^2]$
10	$[(s + 0.0279)^2 + 1.0033^2][(s + 0.0810)^2 + 0.9051^2][(s + 0.1261)^2 + 0.7183^2]$ $\times[(s + 0.1589)^2 + 0.4612^2][(s + 0.1761)^2 + 0.1589^2]$
	$\alpha_p = 1$ dB ($\epsilon = 0.5080$)
1	$s + 1.965$
2	$s^2 + 1.098s + 1.103$
3	$(s + 0.494)(s^2 + 0.490s + 0.994)$
4	$(s^2 + 0.279s + 0.987)(s^2 + 0.674s + 0.279)$
5	$(s + 0.2895)[(s + 0.0895)^2 + 0.9901^2][(s + 0.2342)^2 + 0.6119^2]$
6	$[(s + 0.0622)^2 + 0.9934^2][(s + 0.1699)^2 + 0.7272^2][(s + 0.2321)^2 + 0.2662^2]$
7	$(s + 0.2054)[(s + 0.0457)^2 + 0.9953^2][(s + 0.1281)^2 + 0.7982^2][(s + 0.1851)^2 + 0.4429^2]$
8	$[(s + 0.0350)^2 + 0.9965^2][(s + 0.0997)^2 + 0.8448^2][(s + 0.1492)^2 + 0.5644^2]$ $\times[(s + 0.1759)^2 + 0.1982^2]$
9	$(s + 0.1593)[(s + 0.0277)^2 + 0.9972^2][(s + 0.0797)^2 + 0.8769^2][(s + 0.1221)^2 + 0.6509^2]$ $\times[(s + 0.1497)^2 + 0.3463^2]$
10	$[(s + 0.0224)^2 + 0.9978^2][(s + 0.1013)^2 + 0.7143^2][(s + 0.0651)^2 + 0.9001^2]$ $\times[(s + 0.1277)^2 + 0.4586^2][(s + 0.1415)^2 + 0.1580^2]$
	$\alpha_p = 2$ dB ($\epsilon = 0.7648$)
1	$s + 1.308$
2	$s^2 + 0.804s + 0.637$
3	$(s + 0.402)(s^2 + 0.369s + 0.886)$
4	$(s^2 + 0.210s + 0.928)(s^2 + 0.506s + 0.221)$
5	$(s + 0.1283)[(s + 0.0675)^2 + 0.9735^2][(s + 0.1766)^2 + 0.6016^2]$
6	$[(s + 0.0470)^2 + 0.9817^2][(s + 0.1283)^2 + 0.7187^2][(s + 0.1753)^2 + 0.2630^2]$
7	$(s + 0.1553)[(s + 0.0346)^2 + 0.9867^2][(s + 0.0968)^2 + 0.7912^2][(s + 0.1399)^2 + 0.4391^2]$
8	$[(s + 0.0265)^2 + 0.9898^2][(s + 0.0754)^2 + 0.8391^2][(s + 0.1129)^2 + 0.5607^2]$ $\times[(s + 0.1332)^2 + 0.1969^2]$
9	$(s + 0.1206)[(s + 0.0209)^2 + 0.9919^2][(s + 0.0603)^2 + 0.8723^2][(s + 0.0924)^2 + 0.6474^2]$ $\times[(s + 0.1134)^2 + 0.3445^2]$
10	$[(s + 0.0170)^2 + 0.9935^2][(s + 0.0767)^2 + 0.7113^2][(s + 0.0493)^2 + 0.8962^2]$ $\times[(s + 0.0967)^2 + 0.4567^2][(s + 0.1072)^2 + 0.1574^2]$

Table 4.A.4. Biquadratic Factors for Normalized Lowpass Elliptic Filters[a]

$$H(s) = C\frac{N(s)}{D(s)};\ \omega_p = 1$$

n	α_r	Constant C	Numerator $N(s)$	Denominator $D(s)$
			$\alpha_p = 0.5$dB	
			(A) $\omega_r/\omega_p = 1.1$	
2	2.3	$7.6262\ 10^{-1}$	$s^2 + 1.71408$	$s^2 + 0.42987s + 1.38465$
3	8.5	$9.0441\ 10^{-1}$	$s^2 + 1.37031$	$(s^2 + 0.20787s + 1.14221)$ $\times(s + 1.112286)$
4	17.6	$1.3176\ 10^{-1}$	$(s^2 + 1.2909)(s^2 + 4.34993)$	$(s^2 + 0.12448s + 1.04556)$ $\times(s^2 + 0.99508s + 0.74958)$
5	27.2	$1.6244\ 10^{-1}$	$(s^2 + 1.25932)(s^2 + 2.19309)$	$(s^2 + 0.08164s + 1.02258)$ $\times(s^2 + 0.49230s + 0.76553)$ $\times(s + 0.57311)$
			(B) $\omega_r/\omega_p = 1.5$	
2	8.3	$3.8540\ 10^{-1}$	$s^2 + 3.92705$	$s^2 + 1.03153s + 1.60319$
3	21.9	$3.1410\ 10^{-1}$	$s^2 + 2.80601$	$(s^2 + 0.45286s + 1.14917)$ $\times(s + 0.766952)$
4	36.3	$1.5397\ 10^{-2}$	$(s^2 + 2.53555)(s^2 + 12.09931)$	$(s^2 + 0.25496s + 1.06044)$ $\times(s^2 + 0.92001s + 0.47183)$
5	50.6	$1.9197\ 10^{-2}$	$(s^2 + 2.42551)(s^2 + 5.43764)$	$(s^2 + 0.16346s + 1.03189)$ $\times(s^2 + 0.57023s + 0.57601)$ $\times(s + 0.42597)$
			(C) $\omega_r/\omega_p = 2.0$	
2	13.9	$2.0133\ 10^{-1}$	$s^2 + 7.4641$	$s^2 + 1.24504s + 1.59179$
3	31.2	$1.5424\ 10^{-1}$	$s^2 + 5.15321$	$(s^20.53787s + 1.14849)$ $\times(s + 0.69212)$
4	48.6	$3.6987\ 10^{-3}$	$(s^2 + 4.59326)(s^2 + 24.22720)$	$(s^2 + 0.30116s + 1.06258)$ $\times(s^2 + 0.88456s + 0.41032)$
5	66.1	$4.6205\ 10^{-3}$	$(s^2 + 4.36495)(s^2 + 10.56773)$	$(s^2 + 0.19255s + 1.03402)$ $\times(s^2 + 0.58054s + 0.52500)$ $\times(s + 0.392612)$
			$\alpha_p = 1.0$dB	
			(A) $\omega_r/\omega_p = 1.1$	
2	4.02	$6.2911\ 10^{-1}$	$s^2 + 1.71408$	$s^2 + 0.45820s + 1.20993$
3	11.48	6.2086	$10^{-1}\ s^2 + 1.37031$	$(s^2 + 0.19530s + 1.04241)$ $\times(s + 0.81616)$
4	20.83	$9.0870\ 10^{-2}$	$(s^2 + 1.29092)(s^2 + 4.34993)$	$(s^2 + 0.10896s + 1.00968)$ $\times(s^2 + 0.79846s + 0.56704)$
5	30.47	$1.1151\ 10^{-1}$	$(s^2 + 1.25932)(s^2 + 2.19309)$	$(s^2 + 0.06924s + 1.00164)$ $\times(s^2 + 0.40429s + 0.68854)$ $\times(s + 0.44656)$
			(B) $\omega_r/\omega_p = 1.5$	
2	11.19	$2.7562\ 10^{-1}$	$s^2 + 3.92705$	$s^2 + 0.87941s + 1.2144$
3	25.17	$2.1562\ 10^{-1}$	$s^2 + 2.80601$	$(s^2 + 0.37539s + 1.02371)$ $\times(s + 0.59101)$
4	39.5	$1.0570\ 10^{-2}$	$(s^2 + 2.53555)(s^2 + 12.09930)$	$(s^2 + 0.20881s + 0.99881)$ $\times(s^2 + 0.72998s + 0.36428)$
5	53.9	$1.3178\ 10^{-2}$	$(s^2 + 2.42551)(s^2 + 5.43764)$	$(s^2 + 0.13308s + 0.99495)$ $\times(s^2 + 0.45774s + 0.51706)$ $\times(s + 0.33784)$

Table 4.A.4 (*continued*)

n	α_r	Constant C	Numerator $N(s)$	Denominator $D(s)$
			(C) $\omega_r/\omega_p = 2.0$	
2	17.09	$1.3971\ 10^{-1}$	$s^2 + 7.46410$	$s^2 + 0.99894s + 1.17007$
3	34.45	$1.0589\ 10^{-1}$	$s^2 + 5.15320$	$(s^2 + 0.43406s + 1.01059)$ $\times(s + 0.53995)$
4	51.9	$2.5391\ 10^{-3}$	$(s^2 + 4.59326)(s^2 + 24.2271)$	$(s^2 + 0.24295s + 0.99322)$ $\times(s^2 + 0.70254s + 0.31919)$
5	69.36	$4.5634\ 10^{-5}$	$(s^2 + 4.36495)(s^2 + 10.56773)$	$(s^2 + 0.15524s + 0.99190)$ $\times(s^2 + 0.46467s + 0.47186)$ $\times(s + 0.31259)\}$

[a]Note: $\alpha_p = 10 \log (1 + \varepsilon^2)$, $\alpha_r = 10 \log (A^2)$, for ε and A as defined in Figure 4.5. Extensive tables can be found in the literature—e.g., K. L. Su, *Handbook of Tables for Elliptic-Function Filters*, Kluwer Academic Press, Dordrecht (1990).

References

1. L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, New Jersey (1975).
2. A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, New Jersey. (1977).
3. B. Gold and C. M. Rader, *Digital Processing of Signals*, McGraw-Hill, New York (1969).
4. A. Antoniou, *Digital Filters: Analysis and Design*, McGraw-Hill, New York (1979).
5. A. Papoulis, *Signal Analysis*, McGraw-Hill, New York (1977).
6. A. V. Oppenheim, A. S. Willsky, and I. T. Young, *Signals and Systems*, Prentice-Hall, Englewood Cliffs, New Jersey (1983).
7. E. Christian and E. Eisenmann, *Filter Design Tables and Graphs*, Wiley, New York (1966).
8. M. E. Van Valkenburg, *Analog Filter Design*, Holt, Rinehart and Winston, New York (1982).
9. M. S. Ghaussi and K. R. Laker, *Modem Filter Design*, Prentice-Hall, Englewood Cliffs, New Jersey (1981).
10. A. Papoulis, *The Fourier Integral and Its Applications*, McGraw-Hill, New York (1962).
11. R. S. Blum and M. C. Jeruchim, A note on windowing in the simulation of continuous-time communication systems, *IEEE Trans. Commun.* **45**(8), 889–892 (1997).
12. T. W. Parks and J. H. McClellan, Chebyshev approximation of nonrecursive digital filters with linear phase, *IEEE Trans. Circuit Theory* **CT-19**, 189–194 (1972).
13. A. Peled and B. Liu, *Digital Signal Processing: Theory, Design, and Implementation.* Wiley, New York (1976).
14. J. B. Knowles and E. M. Olcayto, Coefficient accuracy and digital filter response, *IEEE Trans. Circuit Theory* **CT-15**, 31–41 (1968).
15. T. Kailath, Channel characterization: Time-variant dispersive channels, in *Lectures on Communication System Theory*, E. J. Baghdady (ed.), McGraw-Hill, New York (1961).
16. R. J. Schwarz and B. Friedland, *Linear Systems*, McGraw-Hill, New York (1965).
17. W. Kaplan, *Operational Methods for Linear Systems*, Addison-Wesley, Reading, Massachusetts (1962).
18. L. Schwartz, *Mathematics for the Physical Sciences*, Addison-Wesley, Reading, Massachusetts (1966).
19. L. A. Zadeh, Frequency analysis of variable networks, *Proc. IRE* **38**, 291–299 (1950).
20. J. B. Cruz and M. E. Van Valkenburg, The synthesis of models for time-varying linear systems. in *Proceedings of the Symposium on Active Networks and Feedback Systems*, Polytechnic Press, Polytechnic Institute of Brooklyn, New York (1960), pp. 527–544.
21. H. D'Angelo, *Linear Time-Varying Systems: Analysis and Synthesis*, Allyn and Bacon, Boston (1970).

This page intentionally left blank

5

Modeling and Simulation of Nonlinear Systems

Perhaps the single greatest justification for the simulation approach to the study of communication systems is the presence of nonlinear elements. In principle, the performance of linear systems can be attacked by analytical means, but the study of nonlinear systems by such means is by and large intractable. Although there have been many analytical studies of one or another nonlinear device, the system context is usually idealized or much simplified compared to realistic scenarios. In addition, it is typical for such analytical formulations to be of such complexity as to require numerical evaluation, a situation which negates the values of analysis—insight and generality. Hence, simulation is generally the appropriate tool for perhaps most nonlinear communication systems because simulation of such systems is no more difficult than for linear systems, given the model. Indeed, the main difficulty in nonlinear system simulation is obtaining the model itself. Hence, much of the focus in this chapter will be on modeling methodology and on associated measuring techniques for determining parameters of models.

In general, a nonlinear system consists of linear and nonlinear elements. Some of the linear elements may be described by linear differential equations, as discussed in the previous chapter, or by other relationships. The nonlinear elements, which are normally relatively few in number, are described by nonlinear functions or nonlinear differential equations relating the input and output of the element. The transform methods described in the previous chapters cannot strictly be applied to nonlinear systems since superposition does not hold in the latter case. The nonlinear system, therefore, generally has to be simulated in the time domain. This applies to the nonlinear part of the system only. The rest of the system can be simulated either in the time domain or the frequency domain (see Figure 5.1), except when it is part of a feedback system. In contrasting linear and nonlinear system simulation, we should make another point. In the linear case, we are generally satified that we will tend to the correct answer under certain limiting conditions (assuming no finite-precision effects), such as letting the sampling interval become smaller and smaller or increasing the duration of a (truncated) finite impulse response. In nonlinear systems, there is no clear set of limiting conditions under which we can assert that the model comes closer to reality, because it is so much more difficult to ascertain the closeness of a nonlinear model to the functioning of the actual device. This reenforces the point made in Chapter 2 about the need to validate simulation models against hardware counterparts.

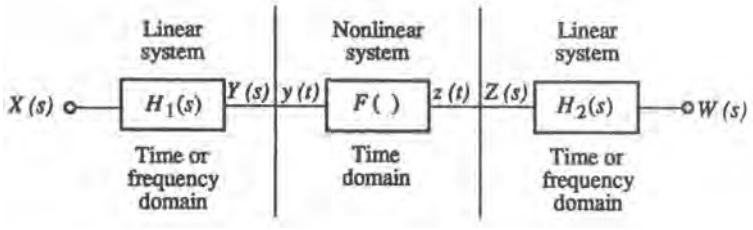


Figure 5.1. Simulation of a nonlinear communication system.

5.1. Modeling Considerations for Nonlinear Systems

Usually devices used in communication systems such as traveling-wave tube amplifiers (TWTAs) and solid-state power amplifiers (SSPAs) are described by analytical models that typically involve solving a set of simultaneous, nonlinear partial differential equations by numerical methods. Unless some suitable simplifications can be made (and this is sometimes possible, as in the optical amplifier discussed later), such detailed models are far too time-consuming to be useful in a system-level simulation where the nonlinear device is just one of many subsystems. Hence, a higher level model is needed, one that converts an input waveform to (nearly) the correct output waveform without necessarily resorting to the fundamental physics of the device. For linear systems, the transfer function is such a model.

In order to speak about nonlinear system models, it will be useful to introduce a number of model-type descriptors:

1. Memoryless models
2. Models with memory
3. Baseband models
4. Bandpass models
5. Block (input/output) models
6. Analytical models
7. Nonlinear differential equation models

These terms capture one aspect or another of the model, but they are not all intended to be mutually exclusive. For example, a model can be both a bandpass and a memoryless model, while its mode of description is an analytical form. Evidently, some of the classifiers *are* mutually exclusive: a model is either of the baseband or bandpass variety, and a model is either memoryless or with memory. We now outline some of these distinctions.

Perhaps the most significant categorical distinction is between a memoryless (or zero-memory) nonlinearity and those with memory. For future reference we shall sometimes abbreviate memoryless models as ZMNL (for zero-memory nonlinearity) and those with memory as NLWM (for nonlinearity with memory). We will also use the term *instantaneous* more or less interchangeably with memoryless. The term "memoryless" implies that the output of a device is a function of the input signal at the present instant only, or perhaps a function of the input at some fixed delay. This implies also that the transfer characteristics are frequency independent. These models are very popular and they fairly accurately characterize a great variety of devices, generally under the condition that the input signal have a bandwidth sufficiently smaller than that of the device. We will discuss memoryless models and different ways of developing them in Section 5.2. In that section we will also examine the primary

consequence of dealing with nonlinearities in the simulation context, which is a required increase in the simulation sampling rate in order to maintain a desired aliasing error.

Memoryless models are an idealization: no physical device is truly (input) frequency independent. Rather, as the bandwidth of an input signal increases, we can expect filtering effects to become more manifest until they are clearly visible. If we allow ourselves only this modeling dichotomy, the decision whether to use a memoryless model or one with memory lies in the art, rather than the science of simulation. Models with memory are considerably more difficult to obtain and, as one would expect, more computationally demanding. Several approaches to developing such models will be elaborated in Section 5.3.

The designations *baseband* and *bandpass* simply refer to the nature of the input signal, although there are generally consequent impacts on the structure of the nonlinearity model itself. An important distinction in this regard is that while the term *bandpass* connotes a modulated carrier, the term *bandpass model* implies one that connects directly the *complex envelopes* of the input and output carriers, as defined in Chapter 3.† This, of course, is the type of model called for if the purpose of the simulation is to extract a performance estimate. Furthermore, in this case, a model that effectively establishes a transfer characteristic for the complex envelope is clearly the most computationally efficient. Not all possible models need be of this type. The term *baseband* model here is intended to connote merely that the input signal's energy is concentrated around zero frequency and we are interested in the nonlinearity's output also in that general vicinity.

Since the goal of communication system simulation is typically to obtain a performance estimate, we are primarily interested in complex envelope models (for bandpass inputs). However, in system design we are also interested in estimating harmonic levels that might be generated, for example, by mixers and limiters. These levels are generally constrained by regulatory agencies. By construction, however, complex envelope models cannot deal with harmonic generation. Different types of models are required for this purpose. While it is not within our intended scope to deal with these models, we shall comment on them at appropriate places.

Returning to our categorization, the descriptors *block models, analytical models*, and *nonlinear differential equations* all basically refer to the structure of models—*how* the models are constructed. Any one of these modalities can be combined with any of the other model descriptors. A block model is one that consists of a set of "blocks" (in the sense of block diagram) connected in series or parallel or a combination thereof. While an analytical model can be presented in block form, the distinction we make here is that the blocks in a block model consist of conventional system elements like filters and memoryless nonlinearities. An analytical model is one based on an analytic construct, which may be, for example, a postulated functional form relating input and output. An analytical model may be memoryless or implicitly with memory: power series are of the first type, while Volterra series are examples of the second. Nonlinear differential equations (NLDEs) are, of course, analytical descriptions, and imply a model with memory. Because NLDEs are an important special case, we shall discuss them separately in Section 5.4.

Virtually any generic type of model will have parameters to be specified or some aspect of its behavior to be determined before it can be applied in simulation to a specific device or class of devices. These undetermined properties must in practice be obtained from measurements which may or may not be simple to make. Some of the model constructions

†Generally, the input and output carrier frequencies need not be the same.

require nonconventional measurements. Hence the practicality of one type of model or another may well hinge on the implied measurements. Thus, in Section 5.5 we spend a little time reviewing these measurement aspects.

Finally, we make the point that we have variously spoken of nonlinear "devices," "blocks," "subsystems," etc. While we are interested in characterizing the most irreducible of these elements, which we usually call a device or a component, the constraints of real hardware may make the terminals available for measurement those of a larger assembly of components. In fact, the methodology associated with constructing models based on measurements, which we will discuss later, does allow us to do so irrespective of what lies between the terminals. Thus, at least in principle, we have the attractive possibility (from a complexity reduction viewpoint) of synthesizing a model for any nonlinear system. However, we might expect that the larger the aggregation between the terminals, the less reliable the model is likely to become.

5.2. Memoryless Nonlinearities

In this section we look at various aspects of modeling memoryless nonlinearities. First we discuss baseband nonlinearities, following which we consider the (generally) bandwidth-expanding effects of a memoryless nonlinearity. In the subsections following, we expose two approaches for developing complex envelope models. The first of these is based on an analytical construction, and the resulting type of model is often referred to as a *bandpass nonlinearity*. The second type of bandpass model we shall refer to as a *bandpass amplifier*, whose modeling evolves from empirical observations. This latter type of model serves as the springboard for some of the models with memory addressed in Section 5.3.

5.2.1. Memoryless Baseband Nonlinearities

The model for a memoryless baseband nonlinearity is characterized by a simple functional relationship of the form

$$y(t) = F[x(t)] \tag{5.2.1}$$

Recall that our definition of a baseband nonlinearity simply means that $x(t)$ is a baseband signal, meaning that its power (energy) is spectrally concentrated around zero frequency, and generally this will also be true of $y(t)$. Although the operator F is independent of that fact, we will for convenience refer to it as a baseband nonlinearity in this context. A relationship such as (5.2.1) is used for most solid-state devices.

Certain nonlinear characteristics can be given in analytical form. An example of such a baseband nonlinearity is a diode. The typical volt–ampere characteristic is

$$I = I_s(e^{\lambda V} - 1) \tag{5.2.2}$$

Here I_s is a device parameter and $\lambda = q/kT$, where q is the charge of an electron, k is Boltzmann's constant, and T is the absolute temperature. The exponential in (5.2.2) can be expanded into a power series, which we can suppose could be truncated after a moderate number of terms with ensuing acceptable error. This suggests that the function F might also be representable by a power series, or by an orthogonal function expansion, or perhaps

reasonably well approximated over the range of $x(t)$ by a polynomial in x. For now we will assume the latter, as it will facilitate later discussion. Thus we can write

$$y(t) = F[x(t)] \approx \sum_{n=0}^{N} a_n x^n(t) \tag{5.2.3}$$

In a practical application, the coefficients a_n would be obtained by fitting a polynomial of degree N to a measured characteristic. It may be that to obtain a good fit would require a large value of N, which would reduce the efficiency of such a model. An alternative, in this case, might be to try a fit using a different type of fitting function. Often, the most efficient and accurate model is to use the raw experimental data themselves in a lookup table, using appropriate interpolation between tabulated data points.

We note that the complex envelope can be interpreted as a (complex) baseband signal. Hence, with an appropriate interpretation of F (which can vary, depending upon the nature of the nonlinearity), most statements that can be made for baseband signals can be suitably carried over to complex envelopes.

5.2.2. Estimating the Sampling Rate for Nonlinear Systems

Given a nonlinear model, the most important implication in simulation is that the sampling rate must be increased. This is because a nonlinearity generally (but not always) increases the bandwidth of a signal, that is, the output bandwidth is greater than the input bandwidth. Since bandwidth is uniquely defined only for a bandlimited signal, let us suppose that the input $x(t)$ is bandlimited to $\pm B/2$. To illustrate the point more easily, let us assume that (5.2.3) holds and that $x(t)$ has finite energy, so that it has a Fourier spectrum $X(f)$. Then, it is easy to see (Problem 5.1) that

$$Y(f) = a_0\delta(f) + \sum_{n=1}^{N} a_n[X(f)^{*^{n-1}} X(f)] \tag{5.2.4}$$

where $^{*^{n-1}}$ indicates $(n-1)$-fold convolution. Hence the term $x^n(t)$ has bandwidth nB (i.e., $\pm\frac{1}{2}nB$).

The situation is not so straightforward for finite-power random processes. But the following result illustrates the theme. For a Gaussian random process $X(t)$ with power spectral density $S_X(f)$ and a large class of nonlinearities, it can be shown[1] that the output PSD is given by

$$S_Y(f) = \sum_{k=1}^{\infty} (b_k^2/\sigma^{2k})[S_X(f)^{*^{k-1}} S_X(f)] \tag{5.2.5}$$

where σ^2 is the average power of $X(t)$ and $\{b_k\}$ is a square-summable sequence of coefficients. Again, if we consider $X(t)$ to be essentially bandlimited to a bandwidth B, the kth term in (5.2.5) has a bandwidth equal to kB.

As we know, the sampling rate for any bandlimited system has to be at least twice the bandwidth in order not to introduce aliasing distortion. Thus, in order to avoid aliasing error for a nonlinear system such as that given by (5.2.3), it would seem that we must have $f_s > 2NB$. This is really overstating the difficulty, however, for two reasons. First, for many

nonlinear systems the coefficients a_n decrease with n. Thus, the contribution of $a_n x^n(t)$ to the energy of $y(t)$ becomes relatively minor for the larger values of n. Hence, the aliasing error in this component alone would make a comparatively smaller contribution to the overall error. Second, the magnitude of the spectrum of $x^n(t)$ is not uniformy distributed over nB. Indeed, as n increases, we can generally expect the spectrum of $x^n(t)$ to be increasingly concentrated around zero frequency and to decrease more rapidly toward zero value near the band edge. Hence, if the sampling rate is less than $2NB$, only the relatively low valued tails of the spectrum are aliased, and this error may well be quite small. The point is that an N-fold increase in sampling rate is almost certainly not required, but *some* increase in sampling rate will be required to keep the aliasing error at an acceptably low level.

Because a realistic nonlinear system will not in general be representable by such a simple form as (5.2.3), or if it could, it would probably not be manifest on the surface, the determination of a proper sampling rate can best be made experimentally, by running some relatively short test simulations with different sampling rates. This, of course, applies not only to the memoryless models discussed here, but to those with memory as well. In fact, at least for the variety of NLWM models that consist of cascades of filters and memoryless nonlinearities (Section 5.3.1, for example), it is clear that the bandwidth expansion takes place in the memoryless nonlinearities, for filters cannot do so.

To anticipate results which will become clear in the next section, we note that the output of a nonlinear system with a bandpass input is more complicated and includes, in addition to spreading of the bandwidth of the signal, spectral components centered around the frequencies $\pm nf_c, n = 0,1,\ldots$ (Figure 5.2), where f_c is the carrier frequency. In the majority of applications in which we are interested the components of the output spectrum center around $\pm f_c$. This partial signal, usually called the first or principal zone output, is obtained by filtering the output with a bandpass filter. This will be discussed in detail shortly.

The increase in the output bandwidth is a phenomenon sometimes referred to as *spectral regrowth*. It was apparently first taken note of in conjunction with satellite applications.[2] In that context, the final power amplification, which was done with a TWT operated into saturation, was seen to raise the level of adjacent channel interference well above the preamplified levels.

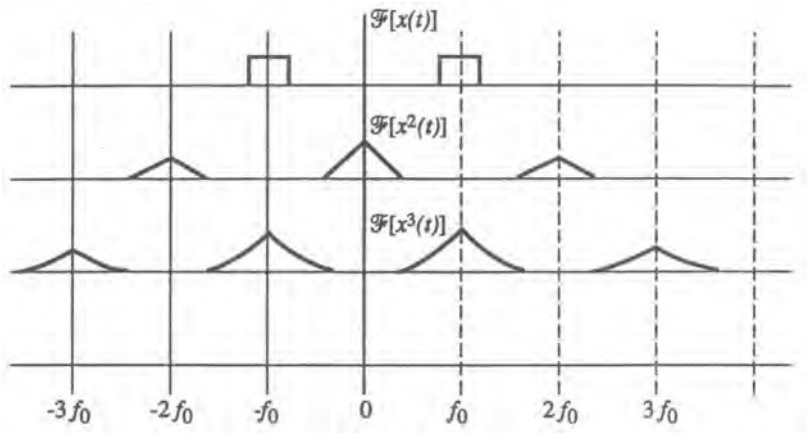


Figure 5.2. Spectral components of the function $x^n(t)$; $\mathcal{F}[\cdot]$ is the Fourier transform.

We add a final note here that nonlinearities do not always increase bandwidth. In fact, they can increase, decrease, or leave unchanged the bandwidth of an input signal, depending on the nonlinearity and the process. A simple example is a random binary waveform input to a square-law device. Here the spectrum collapses to a spectral line at zero frequency.

5.2.3. Memoryless Bandpass Nonlinearities: Analytically Based Models

We define a bandpass nonlinearity as one whose output spectrum is concentrated around a carrier frequency if the input spectrum is itself concentrated around a carrier frequency. A standard model (Figure 5.3) for a bandpass nonlinearity is a memoryless nonlinearity followed by a filter, usually called a *zonal* filter. In much of the work involving nonlinearities one often starts with a postulated instantaneous characteristic which relates the input voltage $x(t)$ to the output voltage $y(t)$, as in (5.2.1), which we repeat here:

$$y(t) = F[x(t)] \tag{5.2.1a}$$

where F is an arbitrary memoryless operator. Consider now an input bandpass signal

$$x(t) = A(t)\cos[2\pi f_c t + \theta(t)] \tag{5.2.6}$$

and make the substitution

$$\alpha = 2\pi f_c t + \theta(t) \tag{5.2.7}$$

so that, considered as a function of α, the nonlinearity output (supressing t for the moment)

$$z = F(A\cos\alpha)$$

is a periodic function and hence can be expanded in a Fourier series

$$z = a_0 + \sum_{k=1}^{\infty}(a_k \cos k\alpha + b_k \sin k\alpha) \tag{5.2.8}$$

Recalling the definition of α, we see that Equation (5.2.8) makes explicit the *harmonics* of the carrier implied by the sketch of Figure 5.2. Hence, although the model so far is somewhat simplistic, we can see the possibility of predicting harmonic levels with a realistic model for F and a model for a filter characteristic at each harmonic kf_c. As will be seen below, the coefficients $\{a_k\}$ and $\{b_k\}$ are generally functions of time.

For performance evaluation, we are usually interested in the *first-zone* output of the memoryless nonlinearity: the first zone is that existing near the band of the operating frequency f_c and is *defined* as the $k = 1$ term in (5.2.8). An implicit assumption in retaining only this term is that the terms centered around kf_c for $k > 1$ are sufficiently removed in

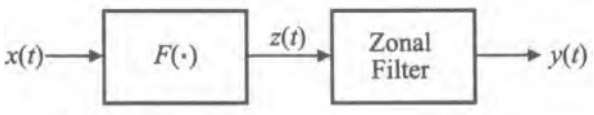


Figure 5.3. Block diagram representation of bandpass nonlinearity.

frequency that their contributed spectral energy to the region around f_c is completely negligible. The coefficients in (5.2.8) are given by the standard formulas, for $k \geq 1$ (we can ignore the dc component in this application),

$$a_k \equiv g_{k1}(A) = \frac{1}{\pi}\int_0^{2\pi} F(A\cos\alpha)\cos k\alpha \, d\alpha \tag{5.2.9a}$$

and

$$b_k \equiv g_{k2}(A) = \frac{1}{\pi}\int_0^{2\pi} F(A\cos\alpha)\ \sin k\alpha \, d\alpha \tag{5.2.9b}$$

For $k = 1$ we will use the abbreviations $g_{11} = g_1$ and $g_{12} = g_2$. Thus, reinserting (5.2.7) and making time explicit, we can write the first-zone output $y(t)$ as

$$y(t) = g_1[A(t)]\cos[2\pi f_c t + \theta(t)] + g_2[A(t)]\ \sin[2\pi f_c t + \theta(t)] \tag{5.2.10}$$

The complex function

$$N(A) = \frac{1}{A}[g_1(A) + jg_2(A)]$$

has long been known in the control systems literature as the describing function.[3] The coefficient $a_1 \equiv g_1$ has also been called the (first-order) *Chebyshev transform* of $F(x)$ by Blachman,[4,5] who also considered the inversion of $g_1(A)$. (See also Problem 5.4.)

Clearly the complex envelope of the first-zone output is given by

$$\tilde{y}(t) = \{g_1[A(t)] - jg_2[A(t)]\}e^{j\theta(t)} \tag{5.2.11}$$

so that, while the full output of the nonlinearity does not have a complex envelope representation, the output of the first zone, and indeed any zone, does have, which makes the use of such a model possible in simulations based on the usual lowpass equivalents.

Equation (5.2.10) implies that the signal can undergo both amplitude and phase distortions. In fact, it can be shown that for any "ordinary" function F the coefficient g_2 will *always* be zero (Problem 5.3). Hence the model under consideration leads to only the first term of (5.2.10),

$$y(t) = g_1[A(t)]\cos[2\pi f_c t + \theta(t)] \tag{5.2.12}$$

and thus can produce only amplitude distortion. Similarly, the complex envelope reduces to the first term of (5.2.11), namely

$$\tilde{y}(t) = g_1[A(t)]e^{j\theta(t)} \tag{5.2.13}$$

which is the form in which it would be implemented in simulation. This is clearly a "time-domain" model, which has a straightforward implementation. Given the (sampled) complex envelope sequence $\{A(kT_s), \theta(kT_s)\}$, the output complex envelope sequence is simply $\{g_1[A(kT_s)], \theta(kT_s)\}$.

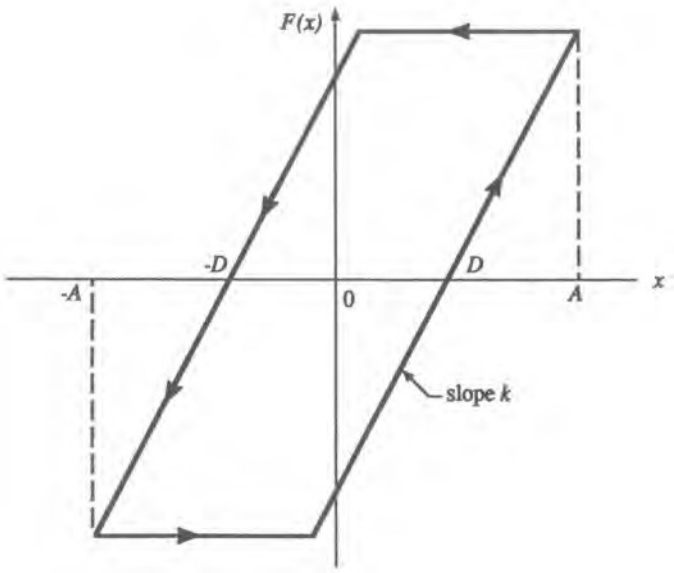


Figure 5.4. A nonlinearity with hysteresis.

In order to retain both terms of (5.2.10) using this general modeling approach, we have to invoke more "unusual" nonlinearities such as the hysteresis characteristic shown in Figure 5.4. Such a nonlinearity might represent certain types of magnetic memories or nonlinear dielectrics. It can be argued that this characteristic cannot properly be labeled "memoryless," though in this instance the term "instantaneous" would be apt. Indeed the notation $F(x)$ when used in conjunction with (5.2.9) must be interpreted in accordance with the arrow convention in Figure 5.4. That is, one must "travel" around the loop in a time-dependent manner, which means that over a cycle, the nonlinearity output takes on the different values dictated by the figure. For this particular nonlinearity, one has (Problem 5.6)

$$\frac{g_1(A)}{A} = \frac{k}{\pi}\left\{\frac{\pi}{2} + \sin^{-1}\left(1 - \frac{2D}{A}\right) + 2\left(1 - \frac{2D}{A}\right)\left[\frac{D}{A}\left(1 - \frac{D}{A}\right)\right]^{1/2}\right\} \qquad \textbf{(5.2.14a)}$$

$$\frac{g_2(A)}{A} = \frac{4kD}{\pi A}\left(1 - \frac{D}{A}\right) \qquad \textbf{(5.2.14b)}$$

for $A \geq D$. Many more instances of describing functions can be found in the literature, e.g., Ref. 3.

For arbitrary instantaneous nonlinearity $F(x)$ the envelope transfers g_1 and g_2 are obtained in general from (5.2.9). In specific cases, however, it may be possible to proceed in a more direct fashion to obtain the first-zone output. In the next two subsections we consider first the family of limiter characteristics for which we will have to use (5.2.9) to obtain g_1, and in the subsection following we consider the "power series" model† of (5.2.3), which yields results more directly.

†This term is somewhat of a misnomer, but is appropriately suggestive.

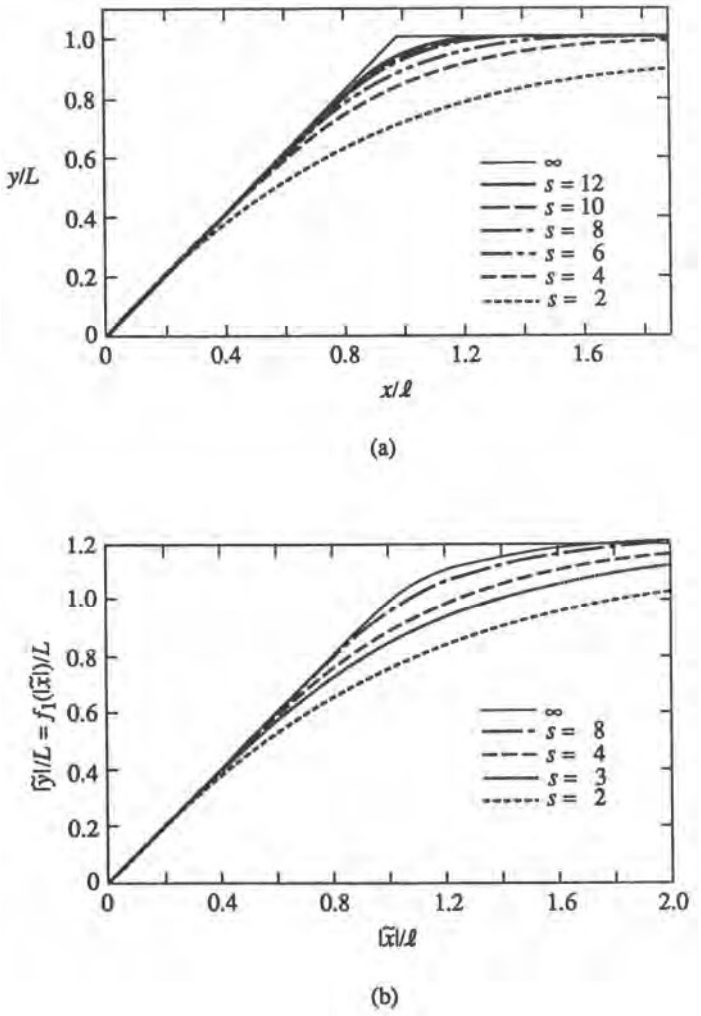


Figure 5.5. Transfer characteristics of lowpass and bandpass limiters. (a) Transfer characteristics of a limiter family; (b) first-zone output for a bandpass limiter. (From Ref. 6, ©IEEE, 1980.)

5.2.3.1. The Limiter Family

A class of memoryless nonlinear devices which can be analyzed using the previous development is the limiter family, members of which are used in a wide variety of communication circuits both at baseband and carrier frequency. The following useful parametric form has been suggested[6] for the memoryless nonlinearity:

$$y = F(x) = \frac{L\ \mathrm{sgn}(x)}{[1 + (l/z)^s]^{1/s}} \tag{5.2.15}$$

This family has a great deal of flexibility due to the parameter s. Several examples are shown in Figure 5.5a. Equation (5.2.15) can be considered to describe a baseband limiter. That is, the

instantaneous output y is as given when x is a baseband input. In (5.2.15) the symbols have the following meaning: $z = |x|$; L is the asymptotic output level as $z \to \infty$, and also serves to scale the output; l is the input "limit" level; and s is the "knee sharpness" parameter. Equation (5.2.15) reduces to familiar forms when the parameters are set to the proper values, as will be seen shortly.

A bandpass limiter consists of a limiter per se, followed by a bandpass (zonal) filter, in accordance with the preceding construct. Hence, with $x(t)$ as in (5.2.6), the output is given by the first term on the right side of (5.2.10). The phase is unaffected, but the output envelope, which is the modulus of the complex envelope, $|\tilde{y}|$, is given by

$$|\tilde{y}| = g_1(|\tilde{x}|) \tag{5.2.16}$$

where $|\tilde{x}|$ is the input envelope [the same as $A(t)$]. Here g_1 is calculated from (5.2.9) with F defined by (5.2.15). For the set of F characteristics shown in Figure 5.5a, the corresponding Chebyshev transforms g_1 are given in Figure 5.5b.

Special Cases. (a) Hard-limiter: Here, one only needs to set $l = 0$; s can be arbitrarily set, but for convenience could be set to unity. The resulting transfer characteristics are (Problem 5.7)

$$y = L\ \mathrm{sgn}(x), \qquad \text{baseband} \tag{5.2.17a}$$

$$|\tilde{y}| = \frac{4L}{\pi}, \qquad \text{bandpass} \tag{5.2.17b}$$

(b) Soft-limiter or limiting amplifier: Here we let $s \to \infty$, which results in

$$\text{baseband} \quad \begin{cases} y = L\ \mathrm{sgn}(x), & z > l \\ y = (L/l)x, & z < l \end{cases} \tag{5.2.18a}$$

$$\text{bandpass} \quad \begin{cases} |\tilde{y}| = g_1(|\tilde{x}|), & |\tilde{x}| > l \\ |\tilde{y}| = (L/l)|\tilde{x}|, & |\tilde{x}| < l \end{cases} \tag{5.2.18b}$$

where g_1 is given graphically in Figure 5.5b and explicitly by the following (Problem 5.10), with $A = |\tilde{x}|$:

$$\frac{g_1(A)}{A} = \frac{2L}{\pi l}\left[\sin^{-1}\frac{l}{A} + \frac{l}{A}\left(1 - \frac{l^2}{A^2}\right)^{1/2}\right], \qquad A \geq l \tag{5.2.19a}$$

$$\frac{g_1(A)}{A} = \frac{L}{l}, \qquad A < l \tag{5.2.19b}$$

It is clear that (5.2.17) are special cases of (5.2.18) as $l \to 0$ in the latter. The particular form (5.2.18) has a sharp knee, but by choosing a finite value of s, we can synthesize a characteristic with rounded knee, as is clear from Figure 5.5. We note that the family (5.2.15) can be used for nonlinearities other than limiters. In particular, the nature of the curves in Figure 5.5 is due to the fact that $s > 1$: other types can be created for $s < 1$. Finally, we note that practical limiters do have some amplitude-to-phase conversion, but to accommodate that behavior, we need a more general model.

5.2.3.2. Power Series Model

As mentioned before, this is a frequently assumed model. It has the nice property of being able to expose relatively clearly the effect that it has on an input signal. Again, we repeat the model, from (5.2.3):

$$y(t) = F[x(t)] \approx \sum_{n=0}^{N} a_n x^n(t)$$

with the same input signal as in (5.2.6). Expressing the latter in terms of the complex envelope

$$\begin{aligned} x(t) &= \mathrm{Re}[\tilde{x}(t)e^{j2\pi f_c t}] \\ &= \tfrac{1}{2}[\tilde{x}(t)e^{j2\pi f_c t} + \tilde{x}^*(t)e^{-j2\pi f_c t}] \end{aligned} \tag{5.2.20}$$

and using the binomial expansion for $x^n(t)$, we obtain

$$\begin{aligned} x^n(t) &= \left\{\tfrac{1}{2}[\tilde{x}(t)e^{j2\pi f_c t} + \tilde{x}^*(t)e^{-j2\pi f_c t}]\right\}^n \\ &= \frac{1}{2^n}\sum_{k=0}^{n}\binom{n}{k}[\tilde{x}(t)]^k[\tilde{x}^*(t)]^{n-k}e^{j2\pi f_c(2k-n)t} \end{aligned} \tag{5.2.21}$$

For the first-zone output we see that only terms where n is odd and $2k - n = \pm 1$ can contribute. The first-zone contribution of (5.2.21) is then

$$x_{1z}^n(t) = \frac{1}{2^{n-1}}\binom{n}{(n+1)/2}|\tilde{x}^*(t)|^{n-1}x(t) \qquad \text{for} \quad n \text{ odd} \tag{5.2.22}$$

Hence the complex envelope of the first-zone component of $y(t)$ is

$$\tilde{y}(t) = \tilde{x}(t)\sum_{m=0}^{(N-1)/2}\frac{a_{2m+1}}{2^{2m}}\binom{2m+1}{m+1}|\tilde{x}^*(t)|^{2m} \tag{5.2.23}$$

The bandpass output signal can now be written (see Problem 5.12)

$$\begin{aligned} y(t) &= \mathrm{Re}[\tilde{y}(t)e^{j2\pi f_c t}] \\ &= \left[\sum_{m=0}^{(N-1)/2}\frac{a_{2m+1}}{2^{2m}}\binom{2m+1}{m+1}|\tilde{x}^*(t)|^{2m+1}\right]\cos[2\pi f_c t + \theta(t)] \\ &= g_1[A(t)]\cos[2\pi f_c t + \theta(t)] \end{aligned} \tag{5.2.24}$$

so that in this case we obtain directly the envelope transfer characteristic. The power series model also permits us to obtain directly the harmonic levels in terms of the coefficients, which then allows us to compute what those coefficients should be in order to control those levels.

5.2.4. Memoryless Bandpass Amplifiers: Empirically Based Models

We now consider the second class of memoryless bandpass nonlinearities, which we referred to as bandpass amplifiers. This terminology is intended to connote a device with an *inherently* passband behavior; that is, only bandpass inputs can produce any measurable output. Nevertheless, spurious signals (harmonics) are also produced, even though the output around the original carrier frequency is the desired signal. Bandpass amplifiers are normally characterized by measuring them, and it is these measurements that form a basis for almost all of the models that will be discussed and that are in use, at least for simulation purposes. In Section 5.5 we will discuss these measurements in more detail.

It is an experimental observation[7] that an input

$$x(t) = A\cos(2\pi f_c t + \theta) \tag{5.2.25}$$

into a bandpass amplifier produces an output of the form

$$y(t) = g(A)\cos[2\pi f_c t + \theta + \Phi(A)] \tag{5.2.26}$$

If we were to plot $g(A)$ and $\Phi(A)$ as a function of A, we would obtain curves more or less similar to those shown in Figure 5.6, depending on the specific device being measured. The (generally) nonlinear gain function $g(A)$ is conventionally referred to as the *AM/AM characteristic* and the (generally) nonlinear phase function $\Phi(A)$ is called the *AM/PM* characteristic, and also often referred to as AM/PM conversion. Sometimes, for brevity, we will refer to both AM/AM and AM/PM as the nonlinear characteristics.

The conditions under which (5.2.26) would be measured are such that A is varied very slowly. This is sometimes called a static or quasistatic measurement. It is natural to ask, therefore, whether the form (5.2.26) can be formally extended if now we modulate the carrier, that is, $A \to A(t)$, such that the bandwidth is now nonzero. We would feel justified doing so if the bandwidth in question is relatively small compared to the inherent bandwidth of the

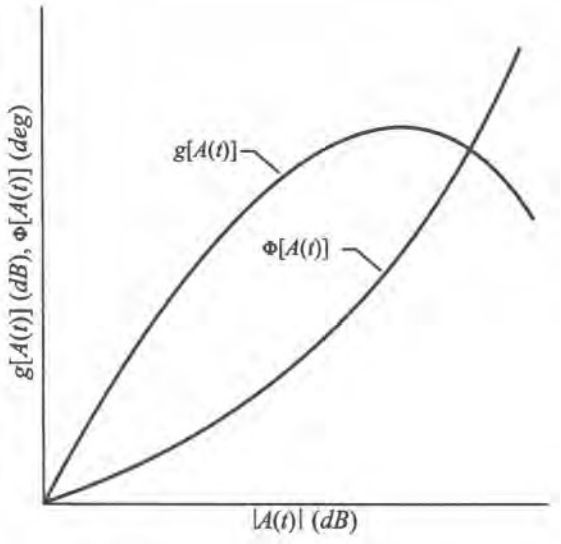


Figure 5.6. Illustration of the amplitude and phase transfer characteristics of an envelope nonlinearity.

device. We can ask the same question as before if we modulate the carrier by letting the phase be time-varying, $\theta \rightarrow \theta(t)$, and again we would answer the same way. Thus, under conditions which can only be qualitatively phrased, we will suppose that the formal extension of (5.2.26) is permitted when the carrier is amplitude- and/or phase-modulated, so that if

$$x(t) = A(t)\cos[2\pi f_c t + \theta(t)] \tag{5.2.27}$$

then

$$y(t) = g[A(t)]\cos\{2\pi f_c t + \theta(t) + \Phi[A(t)]\} \tag{5.2.28}$$

There is experimental evidence[8] that (5.2.28) applies relatively well to solid-state amplifiers, and again we can expect it to be a reasonably good approximation for "narrowband" signals in any device. This model has long been used in analysis and simulation, but begins to fail when the input signal bandwidth becomes large enough. This is when we need to consider the use of a model with memory, which we will discuss in the next section.

We have presented the AM/AM and AM/PM characteristics as quantities which are defined and measured in a definite way. In fact, the way in which one can define (hence measure) these curves is not unique. In Section 5.2.4.3 we will give an example of one variation on the traditional measurement technique that uses a two-tone stimulus. For the moment, however, so long as we retain the memoryless assumption, the model of (5.2.28) applies, no matter how the specific AM/AM and AM/PM curves are obtained.

In terms of our earlier definitions, Equation (5.2.28) is a block model, which can be presented as in Figure 5.7. Figure 5.7a is a symbolic block diagram, while Figure 5.7b is a simulation block diagram showing explicitly what must be done to implement the model in

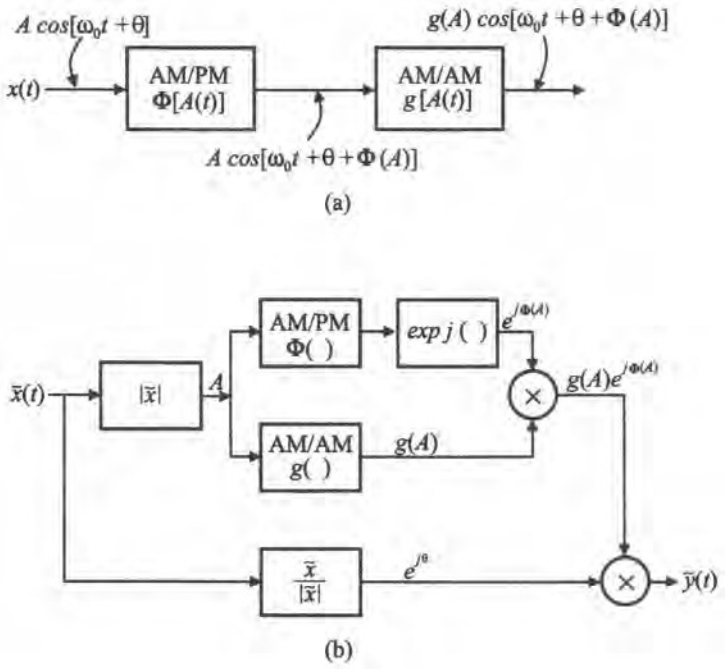


Figure 5.7. Block model for AM/AM and AM/PM envelope (bandpass) nonlinearity; both A and θ are functions of time. (a) Symbolic model at the carrier frequency level; (b) explicit simulation model at the complex envelope level.

code. Because of the structure of Figure 5.7a, we will refer to that particular form as a serial model or, because of the signal representation, as a polar model.

By a standard trigonometric identity, we can expand (5.2.28) into the equivalent form

$$y(t) = S_p(t)\cos[2\pi f_c t + \theta(t)] - S_q(t)\sin[2\pi f_c t + \theta(t)] \tag{5.2.29}$$

where

$$S_p(t) = g[A(t)]\cos\Phi[A(t)] \tag{5.2.30a}$$

and

$$S_q(t) = g[A(t)]\sin\Phi[A(t)] \tag{5.2.30b}$$

In analogy with the previous section, we can identify $S_p(t)$ and $S_q(t)$ with $g_1[A(t)]$ and $g_2[A(t)]$, respectively, so that we can conceive of the amplifier output as the first zone generated by some hypothetical memoryless nonlinearity. The alternative model (5.2.30) is shown in Figure 5.8. Figure 5.8a is again a symbolic representation, while Figure 5.8b shows an explicit structure for purposes of implementing the model. We will refer to this model as a

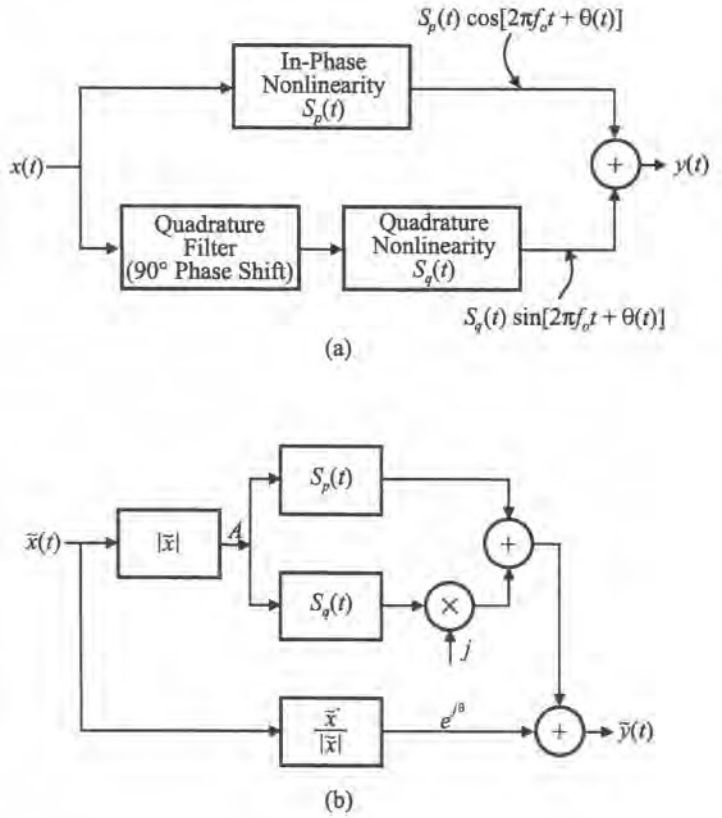


Figure 5.8. Quadrature model of envelope nonlinearity. (a) Symbolic model at the carrier frequency level; (b) explicit model at the complex envelope level.

parallel or quadrature model. At this point the parallel model and the serial model are totally equivalent. However, they do not remain so when they are extended in certain ways to model nonlinearities with memory: this will be seen in the next section.

■ *Remark.* It has been pointed out[9] that a truly memoryless nonlinear system cannot affect the phase, as was already concluded in the previous section. Therefore, the models (5.2.28) and (5.2.29) cannot strictly be called memoryless. However, to make the same distinction as we did in conjunction with the hysteresis nonlinearity in Figure 5.4, we can call the models at hand instantaneous, since they rely only on the present value of the envelope. Following Ref. 9, we can also call this type of model effectively memoryless, which carries the implication that we have already pointed out, that the bandwidth of the input signal is sufficiently smaller than that of the device. A particular structure with memory is considered in Ref. 9 to show how an effectively memoryless model can be extracted under certain limiting conditions (see Problem 5.13). ■

5.2.4.1. *Description and Interpretation of AM/AM and AM/PM Characteristics for Simulation*

Two questions arise with respect to the AM/AM and AM/PM characteristics relative to their use in simulation. One is the proper interpretation of the related measurements into the abstraction which is the simulation model, and the other is simply the manner of description of these functions for simulation use. With respect to the first point, recall that the construction of the nonlinear characteristics is made for different values of A under conditions where each value used can be considered fixed for a long time with respect to all time constants involved. Thus, the average input power is $P_{\text{in}} = A^2/2$ and the average output power is $P_{\text{out}} = g^2(A)/2$. The main point about interpretation of the AM/AM and AM/PM curves in simulation is that while they are obtained under static measurements, they are applied to dynamic situations. That is, we usually interpret $g(\cdot)$ and $\Phi(\cdot)$ as *instantaneous* transfer characteristics, whereas, as just discussed, the corresponding measurements imply an averaging. In fact, the notion of instantaneous, as applied to simulation, must be interpreted to mean no faster than the simulation sampling rate. With this interpretation it can be shown (see Problem 5.10) that the power scale in the measured characteristics can be meaningfully interpreted as *instantaneous power* (a seeming oxymoron!), by which we mean the square of the instantaneous envelope. Thus, to obtain $g(A)$ and $\Phi(A)$ from measurements, we can interpret the power axes as one-half the square of the instantaneous envelope, the latter being the modulus of the complex envelope.

In the typical measurement and display of the nonlinear characteristics, as in Figure 5.6, the abscissa is graduated in units of $10 \log P_{\text{in}}$, and similarly for the output power scale. In order to apply these curves in simulation, the axes with log(power) units must first be converted to numeric units, e.g., via $A = \sqrt{2} \cdot 10^{P(\text{dB})/20}$, where P(dB) is the input or output power in dB units.

The second question relates to how g and Φ are represented. Since these curves are typically obtained experimentally, the most obvious way is simply to construct tables and look up values as the simulation proceeds. Of course, this will necessitate interpolation. Linear interpolation will be adequate if obtained points are sufficiently closely spaced. An alternative is to use an analytical representation. In the absence of measurements we might want to use some such representation which is more or less typical, but even with available measurements we might try to fit these with perhaps an easy-to-evaluate function. If the fit is good, the error will be minimal and the model could thereby be more efficient, and an analytical form is at

least usually helpful for visualization. Many different kinds of forms have been used in the literature. For example, the form

$$\Phi(A) = c_1(1 - e^{c_2 P_{\text{in}}}) + c_3 P_{\text{in}} + \phi_0$$

has been suggested,[7] where the coefficients c_1, c_2, c_3 are determined by a best fit. Because of the exponential, however, a simple table lookup is probably more efficient.

As an illustration of a computationally efficient type of fit, consider the following forms, which have been proposed by Saleh[10]:

$$g(A) = \frac{\alpha_g A}{1 + \beta_g A^2} \tag{5.2.31a}$$

and

$$\Phi(A) = \frac{\alpha_\Phi A^2}{1 + \beta_\Phi A^2} \tag{5.2.31b}$$

for the serial model, and the alternative forms

$$S_p(A) = \frac{\alpha_p A}{1 + \beta_p A^2} \tag{5.2.32a}$$

and

$$S_q(A) = \frac{\alpha_q A^3}{(1 + \beta_q A^2)^2} \tag{5.2.32b}$$

for the parallel model. The coefficients in (5.2.31)–(5.2.32) are determined by a least-squares fit, and it has been shown[10] that the resulting functions provide excellent agreement with several sets of measured data. Still, in terms of computational efficiency there is probably not much to choose between these analytical forms and table lookup. Probably the greatest utility of analytical models is their application to sets of g and Φ measured with frequency as a parameter, which leads naturally to block models for nonlinearities with memory, as will be seen.

5.2.4.2. *Lowpass Equivalent of a Bandpass Amplifier*

The lowpass-equivalent representation of the model we have been discussing is inherent in the modeling construct and evident on the surface. For the serial model, (5.2.28), it is clear that the output complex envelope is given by

$$\tilde{y}(t) = g[A(t)]e^{j\theta(t)}e^{j\Phi(A(t))} \tag{5.2.33a}$$

$$= A(t)e^{j\theta(t)}\frac{g[A(t)]}{A(t)}e^{j\Phi(A(t))} \tag{5.2.33b}$$

$$= \tilde{x}(t)G(|\tilde{x}(t)|) \tag{5.2.33c}$$

with an obvious identification of the complex gain G, which can be interpreted as the lowpass-equivalent transfer function of the nonlinearity. In actuality, the form (5.2.33a) is simpler to calculate, and would be used.

An alternative representation can be extracted for the parallel form (5.2.29), for which it is evident that

$$\tilde{y}(t) = \{S_p[A(t)] + jS_q[A(t)]\}e^{j\theta(t)} \tag{5.2.34a}$$

$$= A(t)e^{j\theta(t)}\left\{\frac{S_p[A(t)]}{A(t)} + j\frac{S_q[A(t)]}{A(t)}\right\} \tag{5.2.34b}$$

$$= \tilde{x}(t)N(|\tilde{x}(t)|) \tag{5.2.34c}$$

where we recognize N as the describing function defined in the preceding section. Again, these alternative forms provide different insights into the action of the nonlinearity, but the form (5.2.34a) would be used in an actual simulation.

5.2.4.3. Alternative Approaches to Defining AM/AM and AM/PM Characteristics

Classically, the AM/AM and AM/PM conversions have been defined with respect to a CW signal as discussed in conjunction with (5.2.25) and (5.2.26). (We will also refer to this kind of characterization as *static*, because the envelope and phase are not time-varying.) However, the definition of these conversions can be generalized in several ways by using *dynamic* rather than static signals.[11] The motivation for this generalization is to improve the accuracy of the resulting model for operational signals that are not necessarily "narrowband," as just discussed. The reasoning here is that because of the natural continuity of the nonlinear device's input/output operator with respect to its input signal, the output predictions of a model constructed from basis signals more complex and "closer" to the operational should be more accurate than those from the classical static model.

Perhaps the simplest basis-signal generalization is that of a two-tone stimulus consisting of a large tone and a small tone at some offset frequency Δf above or below the large tone's frequency f_c (termed *high-side* or *low-side injection*, respectively). In particular, the basis signal can be written in the form of (5.2.27) with (see Problem P5.15)

$$A(t) = A[1 + \alpha^2 + 2\alpha\cos(\pm 2\pi\Delta f t + \theta_a - \theta_A)]^{1/2} \tag{5.2.35a}$$

and

$$\theta(t) = \tan^{-1}\left[\frac{\sin\theta_A + \alpha\sin(\pm 2\pi\Delta f t + \theta_a)}{\cos\theta_A + \alpha\cos(\pm 2\pi\Delta f t + \theta_a)}\right] \tag{5.2.35b}$$

where A and a represent the static amplitudes of the large and small tones, respectively, their corresponding phases θ_A and θ_a are arbitrary, and

$$\alpha \equiv \frac{a}{A} \ll 1 \tag{5.2.36}$$

Condition (5.2.36) is maintained as the amplitude A is swept from a small-signal to a large-signal value that saturates the device being measured. In this case, the dynamic AM/AM and

AM/PM conversions are in general a function of two frequency parameters (f_c and Δf) instead of the single parameter f_c as in the static case. Assuming that the basis signal has its large-tone frequency fixed and only the offset frequency is varied, the output signal can again be represented by expression (5.2.28) with

$$g = g[A(t), \pm\Delta f], \qquad \Phi = \Phi[A(t), \pm\Delta f] \tag{5.2.37}$$

where $A(t)$ and $\theta(t)$ are as in (5.2.35), and the sign on Δf indicates whether high-side or low-side injection is taking place (since, in general, the device's response to these two stimuli need not be identical). By expanding the output (5.2.28) in a McClaurin series with respect to the small parameter α, and assuming high-side injection (low-side injection is entirely similar), it can be shown that, to order α^2 (see Problem P5.16),

$$\begin{aligned} y(t) &= g(A, \Delta f)\cos[2\pi f_c t + \theta_A + \Phi(A, \Delta f)] \\ &\quad + \alpha g(A, \Delta f)\{S_+ \cos(2\pi f_+ t + \Phi_+) - S_- \cos(2\pi f_- t + \Phi_-)\} + O(\alpha^2) \end{aligned} \tag{5.2.38a}$$

where

$$S_\pm \equiv S_\pm(A, \Delta f) = \frac{1}{2}\left\{\left[1 \pm \frac{Ag'(A, \Delta f)}{g(A, \Delta f)}\right]^2 + [A\Phi'(A, \Delta f)]^2\right\}^{1/2} \tag{5.2.38b}$$

are the amplitudes of the *two* sidebands present at the output of the nonlinearity at the frequencies $f_\pm \equiv f_c \pm \Delta f$, with similar expressions for the sideband phase components $\Phi_\pm$. The primes in (5.2.38b) represent derivatives of the dynamic conversions (with $\alpha = 0$) with respect to A. Observe from (5.2.38b) that the differential equations for the dynamic AM/AM and AM/PM conversion curves $g(A, \Delta f)$ and $\Phi(A, \Delta f)$ can be obtained from just the amplitude measurements of the two output sidebands, and these in turn can be solved using the appropriate small-signal conditions. It can be seen that, as developed so far, $\alpha \to 0$ is a necessary condition for the first part of the expression on the right-hand side of (5.2.38a) to be a good approximation for the output. Further discussion of the measurement procedures and model construction involved in this approach can be found in Section 5.5 below. The result of such a construction for a representative satellite communications helix TWT is shown in Figure 5.9. There is clearly a difference between the curves obtained by the method just described and the one conventionally obtained.

By employing the Hilbert transform described in Chapter 3, a dynamic AM/AM and AM/PM conversion can be defined for more general bandpass basis signals, such as a more general two-tone input for which the smallness condition (5.2.36) does not necessarily hold; a triangular-wave AM signal that in essence represents a multitone signal with a distinct set of amplitudes; or a more general, digitally modulated signal such as QAM.

5.2.4.4. Multiple Carriers and Intermodulation Products

Although the preceding development is phrased in terms of an amplitude- and/or phase-modulated input carrier, the lowpass-equivalent model described is equally applicable to multiple modulated carriers (Problem 5.13). This is because the form (5.2.27) can represent one or several carriers at the input to the amplifier. To see this, consider a multiple carrier input

$$x(t) = \sum_{i=1}^{N} A_i(t)\cos[(\omega_c + \omega_i)t + \theta_i(t)]$$

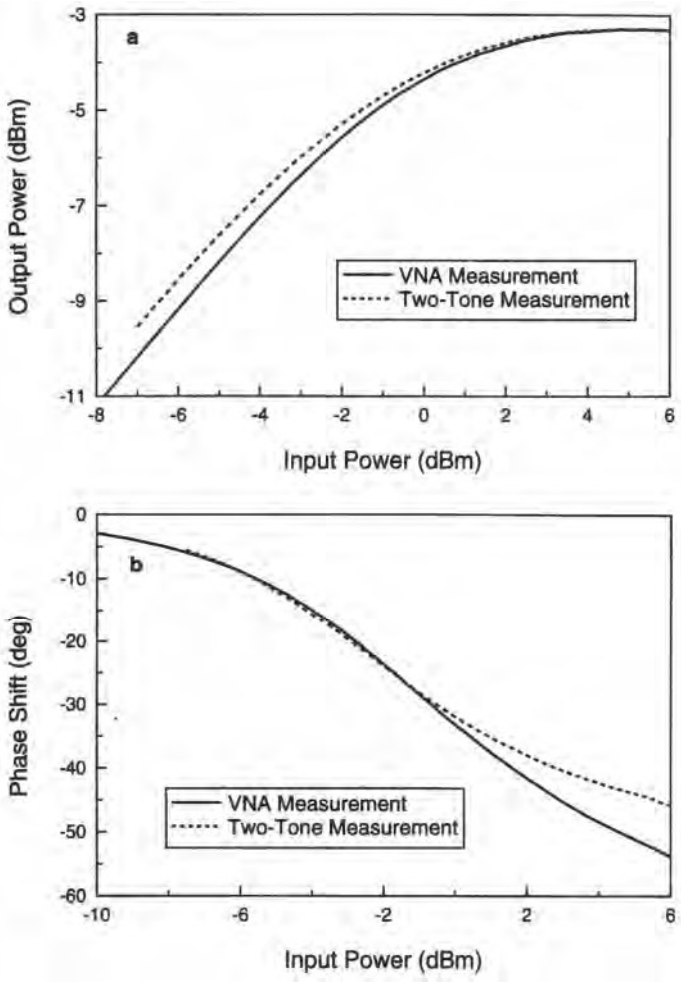


Figure 5.9. Comparison of amplifier characteristics measured with VNA and a two-tone technique. (a) AM/AM characteristic; (b) AM/PM characteristic.

which can be written as

$$x(t) = A(t)\cos[\omega_c t + \theta(t)] \tag{5.2.39}$$

where

$$A(t) = [x_I^2(t) + x_q^2(t)]^{1/2} \tag{5.2.40a}$$

$$\theta(t) = \tan^{-1}[x_I(t)/x_q(t)] \tag{5.2.40b}$$

$$x_I(t) = \sum_{i=1}^{N} A_i(t)\cos[\omega_i t + \theta_i(t)] \tag{5.2.40c}$$

$$x_q(t) = \sum_{i=1}^{N} A_i(t)\sin[\omega_i t + \theta_i(t)] \tag{5.2.40d}$$

Thus, it is clear that the output of the envelope nonlinearity can be expressed in the form (5.2.28) or (5.2.29), where $A(t)$ and $\theta(t)$ are functions of the several input carriers. The multiple carrier case is of obvious importance in multiple access scenarios that use frequency-division multiplexing. Here, *intermodulation* (IM) *products* are a major potential source of degradation, which induce a typical tradeoff between performance (IM product levels) and power efficiency (amplifier backoff). It should be emphasized that the model at hand is a lowpass equivalent, so that the IM products that can be revealed are only those existing in the first zone, i.e., in the general vicinity of the carrier.

We note that this subject has received a great deal of attention in the literature, using various analytic techniques: Refs. 12–16 are just a small sample. For example, one form of the complex envelope of the nonlinearity output is given by[15]

$$\tilde{y}(t) = \sum_{k_1+\cdots+k_N} = 1M(k_1, \ldots, k_N, A_1, \ldots A_N) \exp\left\{ j \sum_{p=1}^{N} k_p[\omega_p t + \theta_p(t)] \right\} \tag{5.2.41}$$

where

$$M = \frac{1}{(2\pi)^2} \int_0^\infty \int_0^\infty u \left[\prod_{p=1}^{N} J_{k_p}(uA_p) \right] J_1(u\rho)\rho g(\rho) e^{j\Phi(\rho)}\, du\, d\rho \tag{5.2.42}$$

and here ρ represents the multicarrier envelope (5.2.40a). Although these equations give a clear qualitative picture of the interaction among the carriers, the net effect upon any one of them is not so obvious and requires numerical evaluation. While this can reasonably be done for the A_i constant, it becomes problematic when the envelopes of the individual carriers are not constant, for then M becomes a function of time. Even if the individual carriers arrive as constant envelope, filtering at the input to the receiver will generally alter that fact. Thus, in this case, a practical alternative is to let the simulation "solve" the problem.

We should also reiterate a point made earlier, namely, that while we have been considering here a memoryless nonlinearity, the discussion applies also to certain models with memory in which (as will be seen) memoryless nonlinearities appear sandwiched by linear filters. For, in this case, intermodulation can be brought about only by the memoryless nonlinearities. We also note that although the bandwidth expansion (hence sampling rate increase) is fundamentally caused by the same mechanism as before, here it manifests itself differently. With respect to the simulation sampling rate, it is the IM products that fall outside the original band covered by the input carriers that need to be anticipated, so that they do not fold back onto the spectrum of interest.

Regarding the sampling rate, aside from the bandwidth expansion just mentioned, it should also be clear that the sampling rate for N equal bandwidth carriers is at least N times that for a single such carrier (including guard bands). This point is discussed further in Chapter 8. Typically, we are interested in the performance of a "typical" carrier, but we must nevertheless account for the presence of all. If indeed we want to concentrate on a typical single carrier, it is possible to reduce the run time to something like that needed if that carrier were by itself. The reader is referred to Ref. 17 for the details.

5.2.4.5. Setting the Operating Point of a Memoryless Nonlinearity

The operating point (OP) of a nonlinear amplifier is the value of average input power that is desired in a particular application. If the input envelope is constant, $A(t) = A$, then the average input power $P_{\text{in}} = A^2/2$ is well defined and the maximum output power would be

extracted by setting the operating point to correspond to the peak of the input/output power curve (usually referred to as saturation).

Let us suppose that there is a preamplifier preceding the nonlinear amplifier itself, and let P_{in} be the average power of the arriving signal $x(t)$ at the input to the preamplifier,

$$P_{\text{in}} = \frac{1}{T}\int_0^T \frac{1}{2}A^2(t)\,dt$$

where T is some averaging period and, as usual, $A(t)$ is the envelope (equivalently, the modulus of the complex envelope). If we want the average input power to the nonlinear amplifier (i.e., the operating point) to be, say, P_o, then we merely need to set the preamplifier gain $G = P_o/P_{\text{in}}$.

We can identify two ways of setting G. In the first case, we might use a calibrating signal like an unmodulated carrier, in which case P_{in} is a constant, irrespective of the averaging interval (which of course cannot be too small). Then G is easily established from the AM/AM curve. Generally, the envelope is time-varying, so that, for a given T, so is P_{in}. In practice, of course, the preamplifier is really an automatic gain control (AGC), which adjusts the gain in accordance with the "current" input power. The latter can be defined as a sliding time average

$$P_{\text{in}}(t) = \frac{1}{T}\int_{t-T}^{t} \frac{1}{2}A^2(t)\,dt$$

so that, with T reasonably defined, the AGC gain is simply $G(t) = P_o/P_{\text{in}}(t)$. Since, in simulation, time is incremented by T_s, the possible update rate of G is evidently some multiple of T_s.

5.3. Nonlinearities with Memory (NLWM)

In the preceding section we discussed models for memoryless (or perhaps instantaneous) nonlinearities. These models represent fairly accurately a variety of devices driven by narrowband inputs, and are commonly used in analysis and simulation. The suitability of a memoryless model for narrowband signals stems from the fact that many radiofrequency amplifiers have indeed a wide bandwidth, and over any relatively small portion of that band the transfer characteristic does look nearly frequency independent about the center of that portion.

When, however, we attempt to send "wideband" signals, by which we mean that the bandwidth of the signal is comparable to the inherent bandwidth of the device (no matter what the actual bandwidth), then we are bound to encounter some frequency-dependent behavior. That such behavior exists is also revealed experimentally through a generalization of Figure 5.6. Suppose that instead of using a single tone at f_c, we measure AM/AM and AM/PM characteristics using a set of tones of different frequencies, spanning the bandwidth of the device. The resulting characteristics might have an appearance generally like that shown in Figure 5.10[†], which would be *prima facie* evidence of frequency dependence. If, on the other hand, all the curves at the different frequencies overlay one another, or nearly so, we would

[†]For the moment we are not concerned whether these curves are "realistic." The main point is that they are not identical.

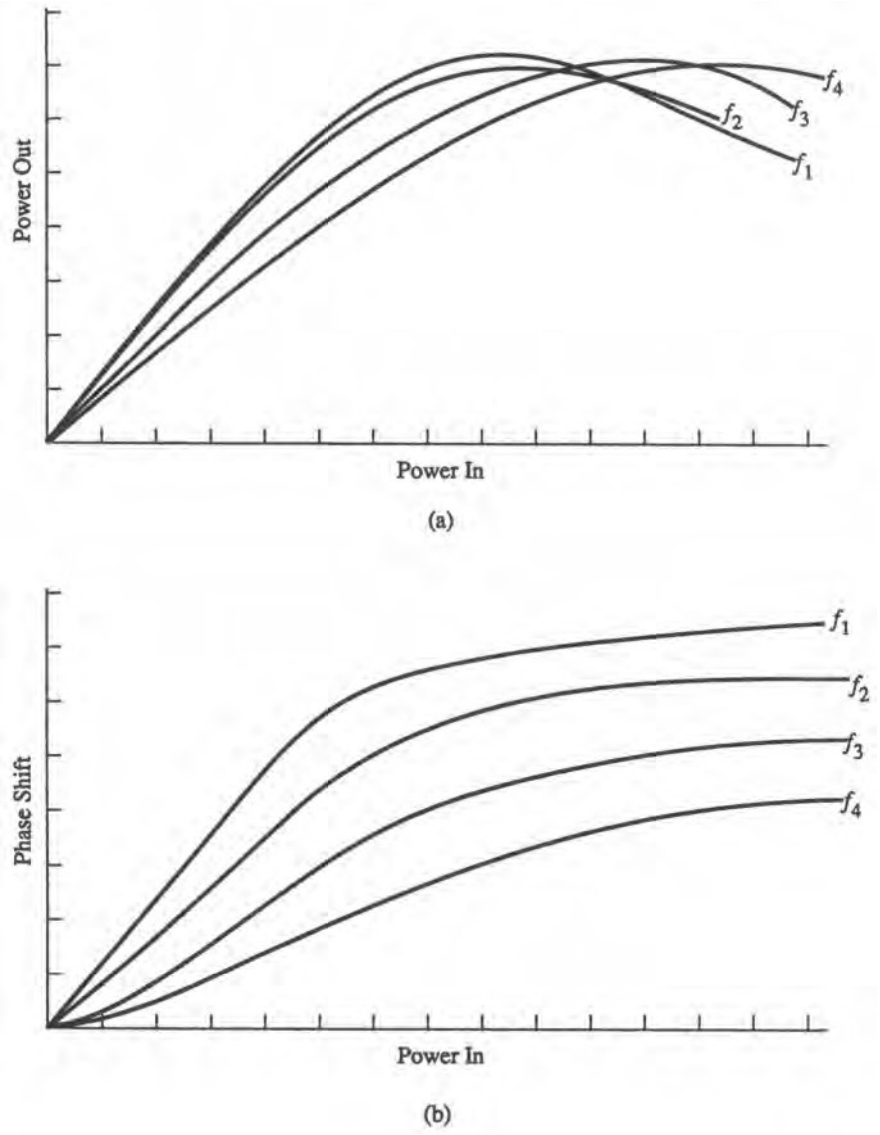


Figure 5.10. Illustration of transfer characteristics of a nonlinear amplifier with memory, as measured by a "static" test (swept-frequency measurements). (a) Gain transfer function with frequency as a parameter; (b) phase transfer function with frequency as a parameter.

have good grounds to believe that a memoryless model would be adequate. Of course, just as with the memoryless model, we are not limited to making measurements with a single tone, and the resulting measured characterization would very likely be different.

In Section 5.1, we began a discussion on the different modeling options for nonlinearities. Here we will continue that discussion focusing only on nonlinearities with memory. As a general statement, a nonlinearity with memory can be defined as one whose output, at any time t, depends on a segment (finite or infinite) of the input signal prior to time t. Memory

implies a filtering effect that is in part manifested by the frequency-dependent nonlinear characteristics mentioned above. It will therefore not be surprising that nonlinear block models will now include blocks that will be identified as filters.

To place NLWM models in perspective, it is useful to define (roughly following Palmer and Lebowitz[18]) three broad classes of characterization: *macroscopic, microscopic*, and *analytical* models. By macroscopic models we mean models constructed on the basis of input/output measurements using an appropriate probing signal. In other words, we treat the nonlinear device (subsystem) as a black box, and use the measurements to "identify" a topology that seems to account for the measurements.† This type of model will typically be a block model, as defined earlier. Because it is the highest level type of modeling, it is computationally the most efficient. Such models also lend themselves best to practical system-level specifications. The drawback to such models is their variable accuracy. Nevertheless, they constitute the standard approach for system simulation, and below we shall concentrate on this type of model.

By microscopic models, we mean models that are *exact* at the intended level of representation. Just as the term macroscopic implies only external access to the box, our use of microscopic implies some degree of visibility inside the box. We can identify three types of microscopic models.

- *Microscopic Models I.* In this modeling construct, subsystems are comprised of some set of well-defined functions operating at the waveform level (for baseband signals) or at the complex envelope level (for bandpass signals). An example of such a subsystem would be one consisting of a cascade of elements some of which are memoryless nonlinearities. If we accept that this set of elements is a perfect representation of the subsystem in question, the model is exact at the *intended* level of representation. Another example of this type of model is a phase-locked loop. The nonlinear differential equation representing the loop is treated as an exact representation of its functioning. This type of model is suitable for communication system simulation.
- *Microscopic Models II.* This type of model applies to subsystems with nonlinear devices containing filtering or energy storage elements for which an equivalent circuit can be developed. By and large, this is the case for solid-state devices.[20] Assuming such circuit equivalents are exact, larger circuits that include these (nonlinear) solid-state devices can be considered exact at the intended level of representation. However, circuit-level models have to be simulated at the actual carrier frequency, making them generally unsuitable for system-level simulation. One way[18] to use circuit-level models is to derive complex envelope properties (such as AM/AM and AM/PM characteristics) and port these results to a system-level waveform simulator. The availability of a circuit model notwithstanding, it may be that the most practical course would be to synthesize a macroscopic type of model.
- *Microscopic Models III.* These include subsystems or devices containing inseparable nonlinear and filtering behavior for which equivalent circuits cannot be readily formulated. The most important example is the traveling-wave tube. Here, the fundamental mode of description is a physical model[21,22] consisting of systems of partial differential equations, which is not suitable for inclusion in a system-level simulation. Here, the usual course is to derive a block model based on black-box characterization.

†An interesting taxonomy of identification has been proposed by Sjöberg *et al.*,[19] who defined "white-," "gray-", and "black"-box modeling, depending upon the degree to which prior knowledge or understanding could be brought to bear in the modeling process. For nonlinear amplifiers, it would probably be fair to say that our procedures fall under the "gray"-box category.

The third general modeling category for NLWM, analytical models, is meant here to refer to models such as the Volterra series that are able, under some circumstances, to represent exactly the functioning of a NLWM. In a sense, such models stand at the intersection of macroscopic and microscopic models because they are both fed by external measurements and at the same time are capable of predicting the device's behavior to the same degree of accuracy as a microscopic model. We shall take a brief look at Volterra series models as well as a less general version, referred to as polyspectral models.

The modeling approaches we shall discuss will be "fed" by one set of measurements or another. The role of measurements is crucial to define the model in a quantitative, not qualitative, sense. That is, the structure of the model is defined independently of the measurements themselves, although a particular modeling idea may well completely specify what measurements must be taken. It may also be that several structures are defined by the same set of measurements, but that these measurements are not used in the same way. For example, the structure may be preconceived according to some heuristic construct, or it may "fall out" from some reasoning process. One or another of the blocks may be directly defined by a particular measurement, or the blocks may be made to fit some set of measurements as best they can within the limitations of that structure. There are many variations on this theme, and in the following we will illustrate some, but by no means all, of the modeling possibilities, which we will group (somewhat loosely) into a few categories.

5.3.1. NLWM Modeling I: Fitting Swept-Tone AM/AM and AM/PM Measurements

In this approach, the measurements are of the type illustrated in Figure 5.10: swept-tone, swept-amplitude measurements are made (Section 5.5) resulting generally in two sets of frequency-dependent curves. The objective in this category of modeling is to construct a block model which, under the same input conditions as the measurements, will reproduce as closely as possible the measured AM/AM and AM/PM curves. While this property of the model is satisfying to some degree, there is no guarantee of closeness for inputs other than single tones. In fact, since superposition does not hold in a nonlinearity, we know that such a model must be an approximation,[23] though it may be a reasonably good one. We note that in this modeling methodology the structure of the model is not predetermined, but emerges as a consequence of the basic objective. We will present three versions of this approach.

5.3.1.1. The Poza–Sarkozy–Berger (PSB) Model [24]

It will be convenient to discuss AM/AM and AM/PM models separately, and then combine the results.

5.3.1.1.1. AM/AM Characteristics. The basis for this model construction is the observation that AM/AM (input/output power) characteristics at different frequencies for TWTs, when displayed in decibel units for both input and output scales, have a very "similar" appearance. While similarity is somewhat subjective, the authors postulated that the similarity was in fact complete. In other words, it was assumed, for the purpose of model construction, that every AM/AM curve could be obtained from any other such curve by some combination of horizontal and vertical displacement. Below, we will refer to this idea as the *fundamental assumption.*

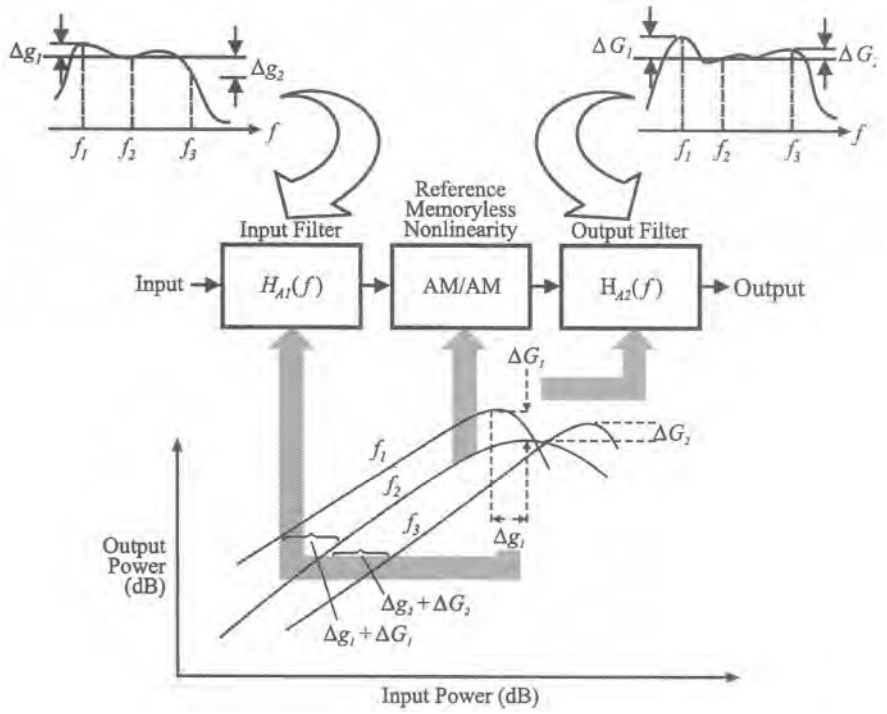


Figure 5.11. The synthesis procedure for the AM/AM portion of the PSB model.

To develop the procedure, we will refer to the simplified sketch in Figure 5.11. Given a set of AM/AM characteristics, we choose one such curve as a *reference* curve; the appropriate one to use is that for the center frequency since this frequency becomes the zero frequency in the lowpass equivalent. In Figure 5.11, the reference curve is the one for $f = f_2$.† In the absence of frequency dependence, that curve would give the input/output relationship for an input sinusoid of any frequency. Suppose now that we input a tone at $f = f_1$. If we were to use the reference curve to project the output level (for a given input level), that output level would be lower than the true value. Hence, if we "preemphasized" the tone at f_1 by the proper amount and then used the reference curve, the corresponding output power would be correct. What is the proper amount of preemphasis? By the fundamental assumption, the curves at f_1 and f_2 differ only by some amount of horizontal and vertical displacement. It is intuitive that the proper amount of horizontal displacement is that needed to align the peaks of the curves, and then the proper amount of vertical displacement is that needed to bring these peaks together. Hence, for each frequency $\neq f_2$ we can interpret the needed amount of horizontal displacement as the required amount of preemphasis, as indicated in Figure 5.11. This, in effect, defines a filter, $H_{A1}(f)$, as shown in the figure (there, the filter's transfer function is defined in the power domain).

If the different AM/AM curves had the same peak power, the input filter $H_{A1}(f)$ would complete the model. However, to account for the difference in the peaks that does exist, we

†For visual clarity, Figure 5.11 shows only three curves. In practice, to get usable resolution we would want measurements at something like 100 frequencies across the band.

now need to postemphasize (or deemphasize) each curve, in accordance with the difference between its peak and that of the reference curve. As shown in Figure 5.11, this effectively generates another filter function, labeled $H_{A2}(f)$, placed after the reference nonlinearity. Thus, to account for the frequency dependence of input/output power, the model consists of a filter-memoryless nonlinearity-filter cascade. In actuality, the fundamental assumption is not quite true, so to define these boxes requires a sophisticated numerical fitting to define in some best way the horizontal and vertical translations.

5.3.1.1.2. AM/PM Characteristics. For these curves, a fundamental assumption analogous to that previously described is supposed to hold, namely, that any one AM/PM curve can be obtained from any other by some combination of vertical and horizontal translation, when the ordinate is measured in degrees (or radians) and the abscissa in decibel units. Again, we eventually have to deal with the fact that this assumption is not quite true and have to construct some associated fitting procedure. But, under the conditions of the fundamental assumption, we can conceive of a procedure similar to that above. Figure 5.12 illustrates the procedure. As before, we take the curve at f_2 as the reference nonlinearity, and construct an input filter that reflects the amount of horizontal movement and an output filter that accounts for the vertical movement. Note that the latter filter has a phase-only transfer function, whereas all other filters are amplitude-only. In the case at hand, the procedure for defining the filters is a little more subtle. The determination of a certain amount of horizontal or vertical displacement requires the identification of a unique feature on the curve with respect to which the displacements can be measured. In the AM/AM case, the saturation point provides such a reference. Typical AM/PM curves, however, such as displayed in Figure 5.12 do not have such a readily obvious feature. One possible procedure to address this problem is to identify a reference point on the basic curve by its slope. Assuming the value of that slope to exist only at one point on the input power axis, we can use it to find the vertical or horizontal movement of the various curves (for different frequencies) with respect to the curve for the reference frequency. This procedure is illustrated in Figure 5.12, where some value of the slope s has been chosen to mark the reference point.

5.3.1.1.3. Combined Model. The final frequency-dependent model is now obtained by cascading the blocks of the two previous figures back to back, as shown in Figure 5.13. The ordering of the blocks follows that previously shown for the memoryless model. Since the connecting blocks are both filters, they can be joined together as one filter, with the proper normalization (Problem 5.14). Although the filters in Figures 5.11 and 5.12 are labeled with actual frequencies $(f_1, f_2, f_3, \ldots)$, it should be clear that their use in the model of Figure 5.13 is as lowpass-equivalent filters, which is accomplished simply by translating the transfer functions from the actual center frequency to zero frequency. It can be seen that the PSB model is an extension of the serial or polar topology defined for memoryless nonlinearities.

It is useful to make the point that while the fundamental assumptions above imply a filter–single memoryless nonlinearity–filter cascade, the converse is also true. That is, a postulated cascade of that sort implies that the resulting swept-tone, swept-amplitude curves will be horizontally and vertically shifted versions of one another (Problem 5.24). This observation has consequences for the properties of certain models.

5.3.1.2. The Saleh Model [10]

This particular model is an extension of the parallel or quadrature topology defined for memoryless nonlinearities. It begins with its own fundamental assumption, that the forms given by (5.2.32a) and (5.2.32b) for a memoryless nonlinearity hold with more or the less the

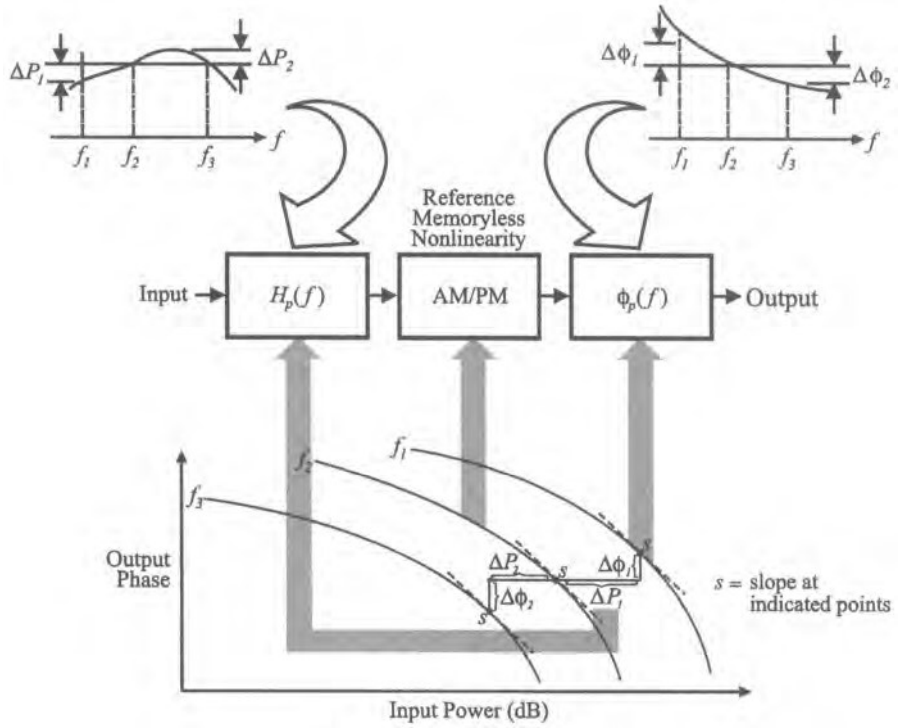


Figure 5.12. The synthesis procedure for the AM/PM portion of the PSB model.

same degree of accuracy for the functions $S_p(A)$ and $S_q(A)$ extracted from the AM/AM and AM/PM curves that apply to any given frequency. If we were to perform the fit implied by (5.2.32a) and (5.2.32b) for any measured nonlinear characteristics at some frequency f, the resulting functions and the fitting coefficients would in general be functions of f. However, it turns out that the rational forms in question provide a better fit if we first remove the "small-signal phase," defined as follows. Let $g(A,f)$ represent any AM/AM curve as a function of A for any given f, and let $\Phi(A,f)$ similarly be any measured AM/PM curve as a function of A for any given f. The small-signal phase characteristic $\Phi_0(f)$ is defined as

$$\Phi_0(f) = \lim_{A \to 0} \Phi[A,f] \tag{5.3.1}$$

Let us return to (5.2.32a) and (5.2.32b) and express modified functions S_p and S_q as functions of (A,f), suppressing the t dependence:

$$S_p(A,f) = g[A,f]\cos\{\Phi[A,f] - \Phi_0(f)\} \tag{5.3.2a}$$

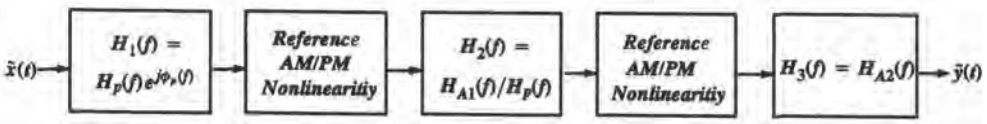


Figure 5.13. The complete PSB model.

and

$$S_q(A,f) = g[A,f]\ \sin\{\Phi[A,f] - \Phi_0(f)\} \tag{5.3.2b}$$

It is these last two equations that are assumed to be well fitted by equations of the form (5.2.32a) and (5.2.32b), respectively, which can now be generalized to

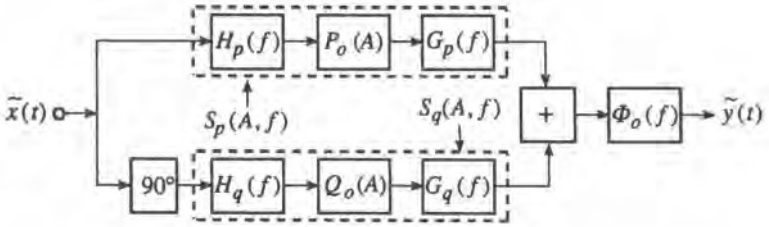
$$S_p(A,f) = \frac{\alpha_p(f)A}{1+\beta_p(f)A^2} \tag{5.3.3a}$$

and

$$S_q(A,f) = \frac{\alpha_q(f)A^3}{[1+\beta_q(f)A^2]^2} \tag{5.3.3b}$$

Equations (5.3.3) are the analytical form of the frequency-dependent model, which can be interpreted as the block model shown in Figure 5.14. The connection between the experimentally derived functions α_p, α_q, β_p, and β_q and the block properties are as given in the legend of Figure 5.14 (Problem 5.22). Notice also that the small-signal phase has been reinserted as an all-pass filter, since it has been removed from the fitting (Problem 5.23).

It is interesting that the block model derived from (5.3.3) consists of (in this case) parallel filter–memoryless amplitude nonlinearity–filter cascades, as was the case with the PSB model. However, the block properties are not the same. Recall the point made earlier that



Legend

The blocks in the figure are defined as follows:

$\Phi_o(f)$ = small signal phase response

$H_p(f) = \sqrt{\beta_p(f)}$ $\qquad G_p(f) = \dfrac{\alpha_p(f)}{\sqrt{\beta_p(f)}}$

$H_q(f) = \sqrt{\beta_q(f)}$ $\qquad G_q(f) = \dfrac{\alpha_q(f)}{\sqrt{\beta_q^{3/2}(f)}}$

$P_o(A) = \dfrac{A}{(1+A^2)}$ $\qquad Q_o(A) = \dfrac{A^3}{(1+A^2)^2}$

Figure 5.14. The frequency-dependent Saleh model (from Ref. 10, @IEEE, 1981).