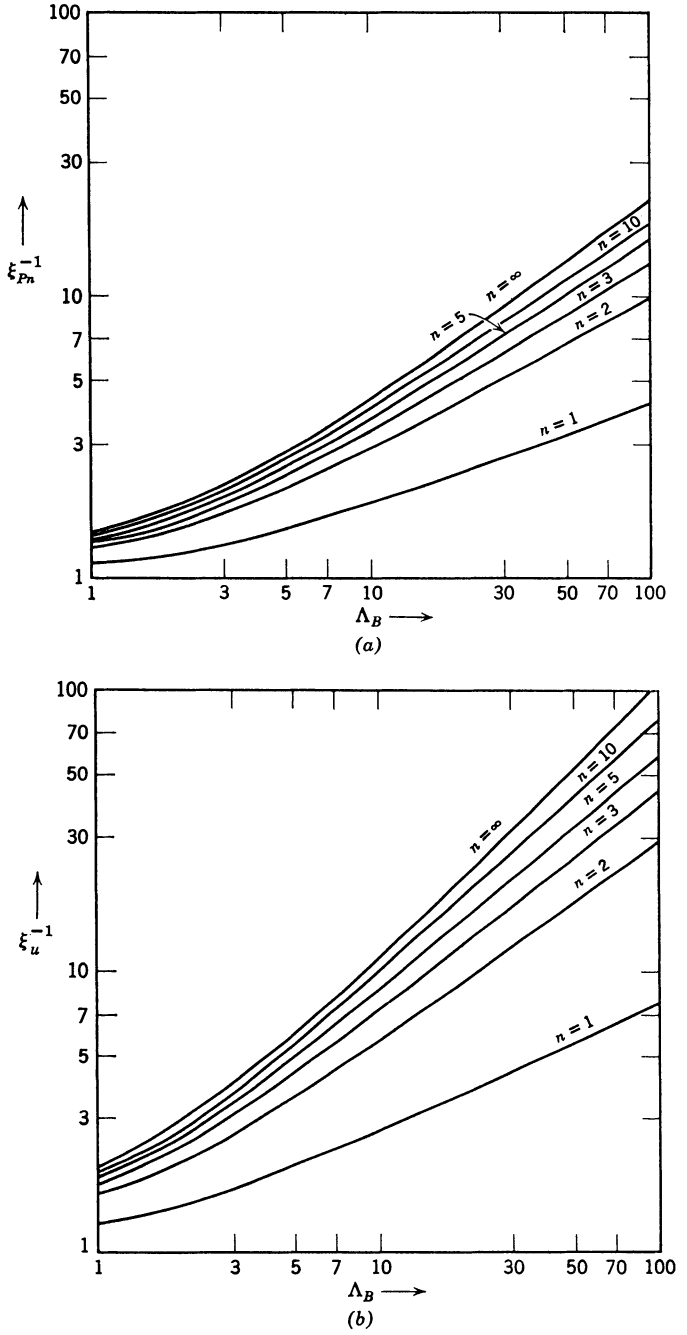


Fig. 6.17 Reciprocal of mean-square error, Butterworth spectra: (a) realizable; (b) unrealizable

Thus the ratio is

$$\frac{\xi_{un}}{\xi_{Pn}} = \frac{\Lambda_B}{1 + \Lambda_B} \frac{1}{\ln (1 + \Lambda_B)}. \tag{162}$$

For $n = \infty$ we achieve appreciable improvement for large Λ_B by allowing delay.

Case 2. Gaussian Family. A second family of spectra is given by

$$S_a(\omega:n) = \frac{2P\sqrt{\pi}\,\Gamma(n)}{k\sqrt{n}\,\Gamma(n - \frac{1}{2})} \frac{1}{(1 + \omega^2/nk^2)^n} \triangleq \frac{d_n}{(1 + \omega^2/nk^2)^n}, \tag{163}$$

obtained by passing white noise through n isolated one-pole filters. In the limit as $n \to \infty$, we have a Gaussian spectrum

$$\lim_{n \to \infty} S_a(\omega:n) = \frac{2\sqrt{\pi}}{k} P e^{-\omega^2/k^2}. \tag{164}$$

The family of Gaussian spectra is shown in Fig. 6.18. Observe that for $n = 1$ the two cases are the same.

The expressions for the two errors of interest are

$$\xi_{Pn} = \frac{N_0}{2P} \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \ln \left[1 + \frac{2d_n/N_0}{(1 + \omega^2/nk^2)^n}\right] \tag{165}$$

and

$$\xi_{un} = \frac{1}{P} \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{d_n}{(1 + \omega^2/nk^2)^n + (2/N_0)d_n}. \tag{166}$$

To evaluate ξ_{Pn} we rewrite (165) in the form of (150). For this case the evaluation of α_i and β_i is straightforward [53]. The results for $n = 1,2,3$, and 5 are shown in Fig. 6.19*a*. For $n = \infty$ the most practical approach is to perform the integration numerically. We evaluate (166) by using a partial fraction expansion. Because we have already found the α_i and β_i, the residues follow easily. The results for $n = 1, 2, 3$, and 5 are shown in Fig. 6.19*b*. For $n = \infty$ the result is obtained numerically. By comparing Figs. 6.17 and Fig. 6.19 we see that the Gaussian spectrum is more difficult to filter than the bandlimited spectrum. Notice that the limiting spectra in both families were nonrational (they were also not factorable).

In this section we have applied some of the closed-form results for the special case of filtering in the presence of additive white noise. We now briefly consider some other related problems.

Colored Noise and Linear Operations. The advantage of the error expression in (152) was its simplicity. As we proceed to more complex noise spectra, the results become more complicated. In almost all cases the error expressions are easier to evaluate than the expression obtained from the conventional Wiener approach.

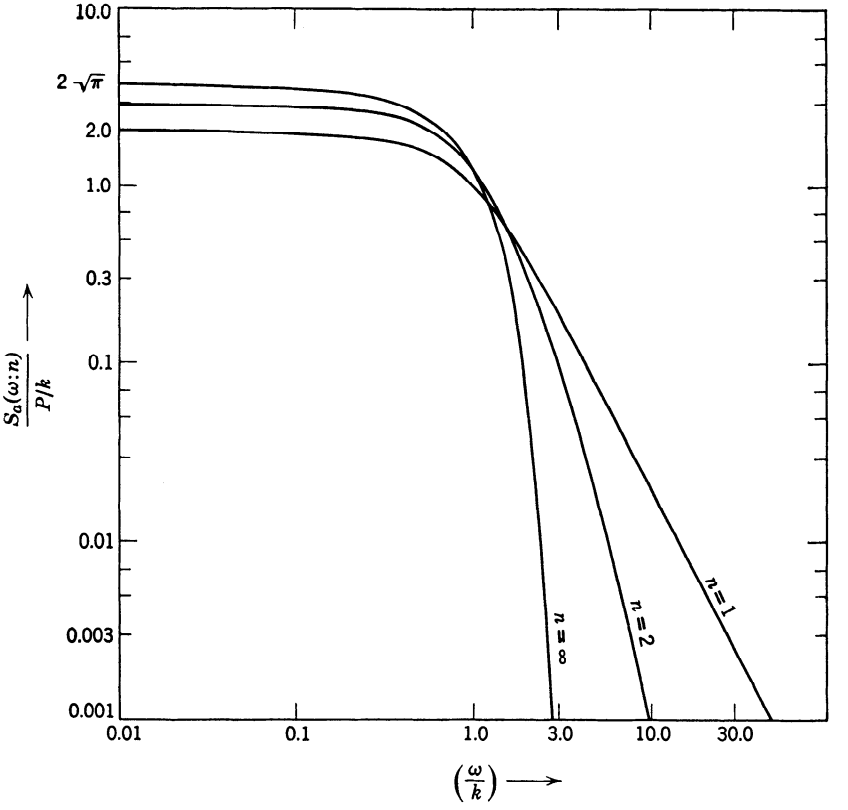


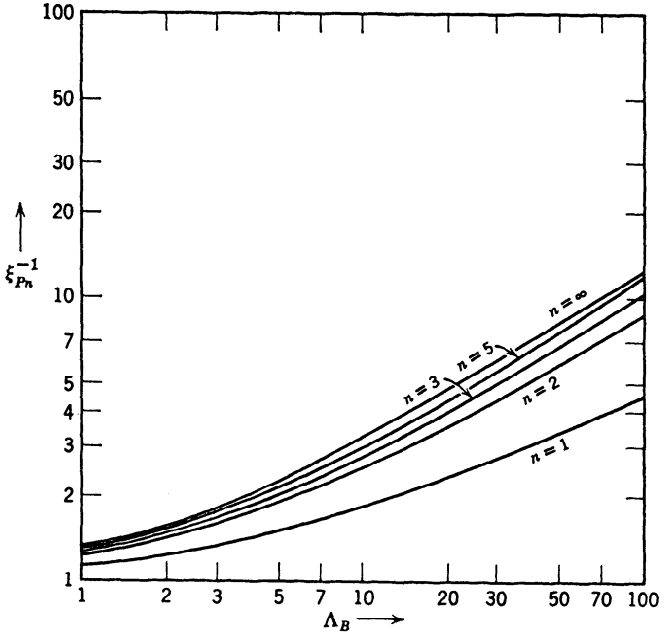
Fig. 6.18 Gaussian family.

For our purposes it is adequate to list a number of cases for which answers have been obtained. The derivations for some of the results are contained in the problems (see Problems 6.2.20 to 6.2.26).

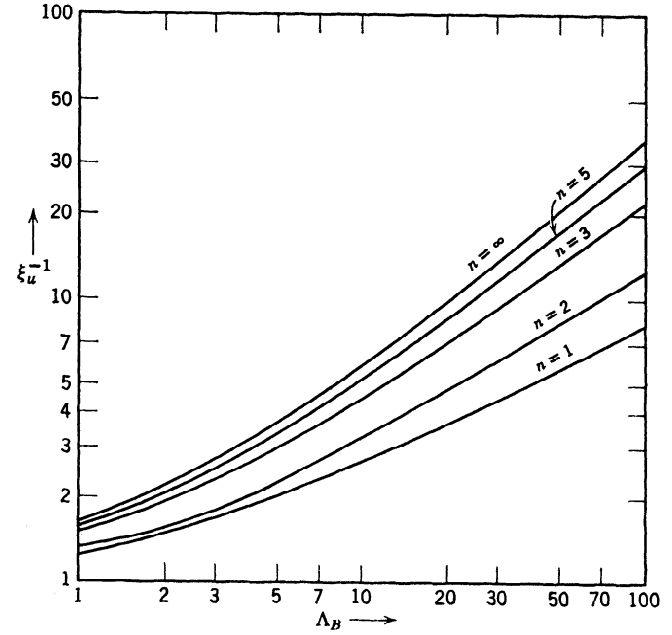
1. The message $a(t)$ is transmitted. The additive noise has a spectrum that contains poles but not zeros.
2. The message $a(t)$ is passed through a linear operation whose transfer function contains only zeros before transmission. The additive noise is white.
3. The message $a(t)$ is transmitted. The noise has a polynomial spectrum

$$S_n(\omega) = N_0 + N_2\omega^2 + N_4\omega^4 + \cdots + N_{2n}\omega^{2n}.$$

4. The message $a(t)$ is passed through a linear operation whose transfer function contains *poles* only. The noise is white.



(a)



(b)

Fig. 6.19 Reciprocal of mean-square error: Gaussian family, (a) realizable; (b) unrealizable.

We observe that Cases 2 and 4 will lead to the same error expression as Cases 1 and 3, respectively.

To give an indication of the form of the answer we quote the result for typical problems from Cases 1 and 4.

Example (from Case 1). Let the additive noise $n(t)$ be uncorrelated with the message and have a spectrum,

$$S_n(\omega) = \frac{2c\sigma_n^2}{\omega^2 + c^2}. \tag{167}$$

Then

$$\xi_P = \sigma_n^2 \exp\left(-\frac{1}{\sigma_n^2}\int_{-\infty}^{\infty} S_n(\omega) \ln\left[1 + \frac{S_a(\omega)}{S_n(\omega)}\right]\frac{d\omega}{2\pi}\right). \tag{168}$$

This result is derived in Problem 6.2.20.

Example (from Case 4). The message $a(t)$ is integrated before being transmitted.

$$r(t) = \int_{-\infty}^{t} a(u)\,du + w(t). \tag{169}$$

Then

$$\xi_P = \frac{N_0}{6} I_1^3 + I_2, \tag{170}$$

where

$$I_1 = \int_{-\infty}^{\infty} \ln\left[1 + \frac{2S_a(\omega)}{\omega^2 N_0}\right]\frac{d\omega}{2\pi}, \tag{171}$$

$$I_2 = \frac{N_0}{2}\int_{-\infty}^{\infty} \omega^2 \ln\left[1 + \frac{2S_a(\omega)}{\omega^2 N_0}\right]\frac{d\omega}{2\pi}. \tag{172}$$

This result is derived in Problem 6.2.25.

It is worthwhile to point out that the *form* of the error expression depends only on the *form* of the noise spectrum or the linear operation. This allows us to vary the message spectrum and study the effects in an easy fashion.

As a final topic in our discussion of Wiener filtering, we consider optimum feedback systems.

6.2.5 Optimum Feedback Systems

One of the forms in which optimum linear filters are encountered in the sequel is as components in a feedback system. The modification of our results to include this case is straightforward.

We presume that the assumptions outlined at the beginning of Section 6.2 (pp. 481–482) are valid. In addition, we *require* the linear processor to have the form shown in Fig. 6.20. Here $g_l(\tau)$ is a linear filter. We are allowed to choose $g_l(\tau)$ to obtain the best $d(t)$.

System constraints of this kind develop naturally in the control system context (see [9]). In Chapter II.2 we shall see how they arise as linearized

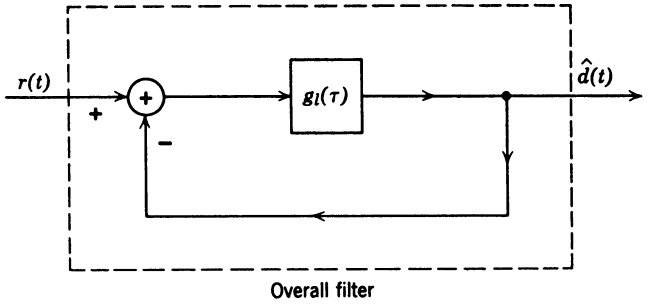


Fig. 6.20 Feedback system.

versions of demodulators. Clearly, we want the *closed* loop transfer function to equal $H_o(j\omega)$. We denote the loop filter that accomplishes this as $G_{lo}(j\omega)$. Now,

$$H_o(j\omega) = \frac{G_{lo}(j\omega)}{1 + G_{lo}(j\omega)} \tag{173}$$

Solving for $G_{lo}(j\omega)$,

$$G_{lo}(j\omega) = \frac{H_o(j\omega)}{1 - H_o(j\omega)}. \tag{174}$$

For the general case we evaluate $H_o(j\omega)$ by using (78) and substitute the result into (174).

For the special case in Section 6.2.4, we may write the answer directly. Substituting (141) into (174), we have

$$\boxed{G_{lo}(j\omega) = \left\{\left(\frac{2}{N_0}\right)^{1/2}\left[S_a(j\omega) + \frac{N_0}{2}\right]^{+} - 1\right\}.} \tag{175}$$

We observe that $G_{lo}(j\omega)$ has the same *poles* as $G^{+}(j\omega)$ and is therefore a stable, realizable filter. We also observe that the poles of $G^{+}(j\omega)$ (and therefore the loop filter) are just the left-half-s-plane poles of the *message* spectrum.

We observe that the message can be visualized as the output of a linear filter when the input is white noise. The general rational case is shown in Fig. 6.21*a*. We control the power by adjusting the spectral height of $u(t)$:

$$E[u(t)\, u(\tau)] \triangleq q\, \delta(t - \tau). \tag{176}$$

The message spectrum is

$$S_a(\omega) = q\left|\frac{b_{n-1}(j\omega)^{n-1} + \cdots + b_0}{(j\omega)^n + p_{n-1}(j\omega)^{n-1} + \cdots + p_0}\right|^2. \tag{177}$$

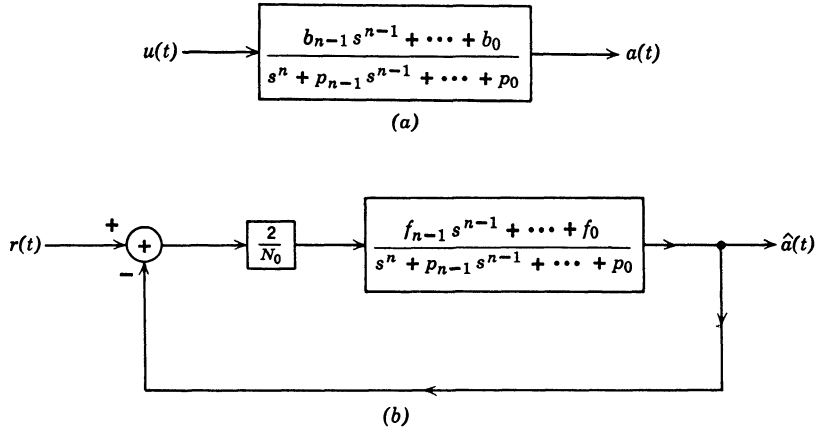


Fig. 6.21 Filters: (*a*) message generation; (*b*) canonic feedback filter.

The numerator must be at least one degree less than the denominator to satisfy the finite power assumption in Section 6.2.4. From (175) we see that the optimum loop filter has the same poles as the filter that could be used to generate the message. Therefore the loop filter has the form shown in Fig. 6.21*b*. It is straightforward to verify that the numerator of the loop filter is *exactly* one degree less than the denominator (see Problem 6.2.27). We refer to the structure in Fig. 6.21*b* as the canonic feedback realization of the optimum filter for rational message spectra [21]. In Section 6.3 we shall find that a general canonic feedback realization can be derived by for nonstationary processes and finite observation intervals.

Observe that to find the numerator we must still perform a factoring operation (see Problem 6.2.27). We can also show that the first coefficient in the numerator is $2\xi_P/N_0$ (see Problem 6.2.28).

A final question about feedback realizations of optimum linear filters concerns unrealizable filters. Because we have seen in Sections 6.2.2 and 6.2.3 [(108*b*) and (127)] that using an unrealizable filter (or allowing delay) always improves the performance, we should like to make provision for it in the feedback configuration. Previously we approximated unrealizable filters by allowing delay. Looking at Fig. 6.20, we see that this would not work in the case of $g_l(\tau)$ because its output is fed back in real time to become part of its input.

If we are willing to allow a postloop filter, as shown in Fig. 6.22, we can consider unrealizable operations. There is no difficulty with delay in the postloop filter because its output is *not* used in any other part of the system.

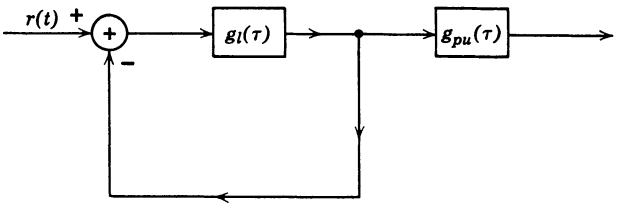


Fig. 6.22 Unrealizable postloop filter.

The expression for the optimum unrealizable postloop filter $G_{puo}(j\omega)$ follows easily. Because the cascade of the two systems must correspond to $H_{ou}(j\omega)$ and the closed loop to $H_o(j\omega)$, it follows that

$$G_{puo}(j\omega) = \frac{H_{ou}(j\omega)}{H_o(j\omega)}. \tag{178}$$

The resulting transfer function can be approximated arbitrarily closely by allowing delay.

6.2.6 Comments

In this section we discuss briefly some aspects of linear processing that are of interest and summarize our results.

Related Topics. *Multidimensional Problem.* Although we formulated the vector problem in Section 6.1, we have considered only the solution technique for the scalar problem. For the *unrealizable* case the extension to the vector problem is trivial. For the realizable case in which the message or the desired signal is a vector and the received waveform is a scalar the solution technique is an obvious modification of the scalar technique. For the realizable case in which the received signal is a vector, the technique becomes quite complex. Wiener outlined a solution in [1] which is quite tedious. In an alternate approach we factor the input spectral matrix. Techniques are discussed in [10] through [19].

Nonrational Spectra. We have confined our discussion to rational spectra. For nonrational spectra we indicated that we could use a rational approximation. A direct factorization is not always possible.

We can show that a necessary and sufficient condition for factorability is that the integral

$$\int_{-\infty}^{\infty} \left| \frac{\log S_r(\omega)}{1 + (\omega/2\pi)^2} \right| d\omega$$

must converge. Here $S_r(\omega)$ is the spectral density of the entire received waveform. This condition is derived and the implications are discussed in [1]. It is referred to as the *Paley-Wiener criterion.*

If this condition is not satisfied, $r(t)$ is termed a deterministic waveform. The adjective deterministic is used because we can *predict* the future of $r(t)$ exactly by using a *linear* operation on only the past data. A simple example of a deterministic waveform is given in Problem 6.2.39.

Both limiting message spectra in the examples in Section 6.2.4 were deterministic. This means, if the noise *were* zero, we would be able to predict the future of the message exactly. We can study this behavior easily by choosing some arbitrary prediction time α and looking at the prediction error as the index $n \to \infty$. For an arbitrary α we can make the mean-square prediction error less than any positive number by letting n become sufficiently large (see Problem 6.2.41).

In almost all cases the spectra of interest to us will correspond to nondeterministic waveforms. In particular, inclusion of white noise in $r(t)$ guarantees factorability.

Sensitivity. In the detection and parameter estimation areas we discussed the importance of investigating how sensitive the performance of the optimum system was with respect to the detailed assumptions of the model. Obviously, sensitivity is also important in linear modulation. In any particular case the technique for investigating the sensitivity is straightforward. Several interesting cases are discussed in the problems (6.2.31–6.2.33).

In the scalar case most problems are insensitive to the detailed assumptions. In the vector case we must exercise more care.

As before, a general statement is not too useful. The important point to re-emphasize is that we must always check the sensitivity.

Colored and White Noise. When trying to estimate the message $a(t)$ in the presence of noise containing both white and colored components, there is an interesting interpretation of the optimum filter.

Let

$$r(t) = a(t) + n_c(t) + w(t), \tag{179}$$

and

$$d(t) = a(t). \tag{180}$$

Now, it is clear that *if* we *knew* $n_c(t)$ the optimum processor would be that shown in Fig. 6.23*a*. Here $h_o(\tau)$ is just the optimum filter for $r'(t) = a(t) + w(t)$, which we have found before.

We do not know $n_c(t)$ because it is a sample function from a random process. A logical approach would be to estimate $n_c(t)$, subtract the estimate from $r(t)$, and pass the result through $h_o(\tau)$, as shown in Fig. 6.23*b*.

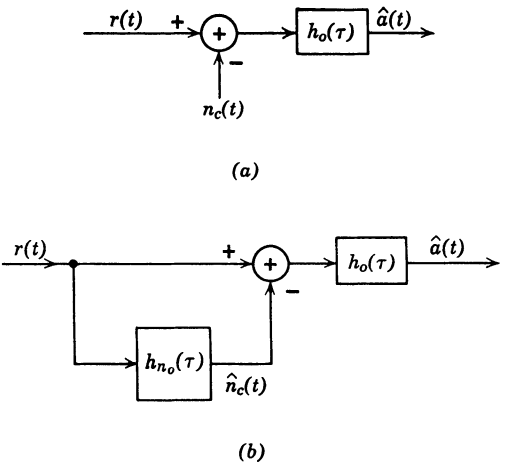


Fig. 6.23 Noise estimation.

We can show that the optimum system does exactly this (see Problem 6.2.34). (Note that $\hat{a}(t)$ and $\hat{n}_c(t)$ are coupled.) This is the same kind of intuitively pleasing result that we encountered in the detection theory area. The optimum processor does exactly what we would do if the disturbances were known exactly, only it uses estimates.

Linear Operations and Filters. In Fig. 6.1 we showed a *typical* estimation problem. With the assumptions in this section, it reduces to the problem shown in Fig. 6.24. The general results in (78), (119), and (122) are applicable. Because most interesting problems fall into the model in Fig. 6.24, it is worthwhile to state our results in a form that exhibits the effects of $k_d(\tau)$ and $k_f(\tau)$ explicitly. The desired relations for uncorrelated message and noise are

$$H_o(j\omega) = \frac{1}{[S_a(\omega)|K_f(j\omega)|^2 + S_n(\omega)]^+} \left[\frac{K_d(j\omega)\, S_a(\omega)\, K_f^*(j\omega)}{[S_a(\omega)|K_f(j\omega)|^2 + S_n(\omega)]^-}\right]_+ \tag{181}$$

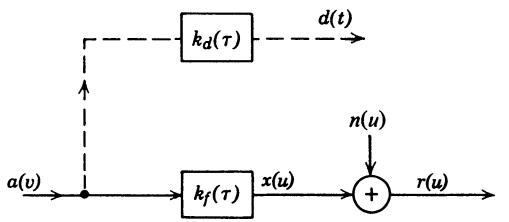


Fig. 6.24 Typical unmodulated estimation problem.

and

$$H_{ou}(j\omega) = \frac{K_d(j\omega)\, S_a(\omega)\, K_f^*(j\omega)}{S_a(\omega)|K_f(j\omega)|^2 + S_n(\omega)}. \tag{182}$$

In the realizable filtering case we must use (181) whenever $d(t) \neq a(t)$. By a simple counterexample (Problem 6.2.38) we can show that linear filtering and optimum realizable estimators *do not* commute in general. In other words, $\hat{d}(t)$ does not necessarily equal

$$\int_{-\infty}^{\infty} k_d(t - \tau)\, \hat{a}(\tau)\, d\tau$$

This result is in contrast to that obtained for MAP interval estimators. On the other hand, comparing (119) with (182), we see that linear filtering and optimum unrealizable ($T_f = \infty$) estimators *do* commute. The resulting error in the unrealizable case when $K_d(j\omega) = 1$ is

$$\xi_{uo} = \int_{-\infty}^{\infty} \frac{S_a(\omega)\, S_n(\omega)}{S_a(\omega)|K_f(j\omega)|^2 + S_n(\omega)} \frac{d\omega}{2\pi}. \tag{183}$$

This expression is obvious. For other $K_d(j\omega)$ see Problem 6.2.35. Some implications of (183) with respect to the prefiltering problem are discussed in Problem 6.2.36.

Remember that we assumed that $k_d(\tau)$ represents an "allowable" operation in the mean-square sense; for example, if the desired operation is a differentiation, we assume that $a(t)$ is a mean-square differentiable process (see Problem 6.2.37 for an example of possible difficulties when this assumption is not true).

Summary. We have discussed in some detail the problem of linear processing for stationary processes when the infinite past was available. The principal results are the following:

1. A constructive solution to the problem is given in (78):

$$H_o(j\omega) = \frac{1}{G^+(j\omega)} \left[\frac{S_{dr}(j\omega)}{[G^+(j\omega)]^*}\right]_+. \tag{78}$$

2. The effect of delay or prediction on the resulting error in an optimum linear processor is given in (108*b*). In all cases there is a monotone improvement as more delay is allowed. In many cases the improvement is sufficient to justify the resulting complexity.

3. The importance of the idea that an unrealizable filter can be approximated arbitrarily closely by allowing a processing delay. The advantage of the unrealizable concept is that the answer can almost always be easily obtained and represents a lower bound on the MMSE in any system.

4. A closed-form expression for the error in the presence of white noise is given in (152),

$$\xi_P = \frac{N_0}{2} \int_{-\infty}^{\infty} \ln \left[1 + \frac{S_a(\omega)}{N_0/2}\right] \frac{d\omega}{2\pi}. \tag{152}$$

5. The canonic filter structure for white noise shown in Fig. 6.21 enables us to relate the complexity of the optimum filter to the complexity of the message spectra by inspection.

We now consider another approach to the point estimation problem.

6.3 KALMAN-BUCY FILTERS

Once again the basic problem of interest is to operate on a received waveform $r(u)$, $T_i \leq u \leq t$, to obtain a minimum mean-square error *point* estimate of some desired waveform $d(t)$. In a simple scalar case the received waveform is

$$r(u) = c(u)\, a(u) + n(u), \qquad T_i \leq u \leq t, \tag{184}$$

where $a(t)$ and $n(t)$ are zero-mean random processes with covariance functions $K_a(t, u)$ and $(N_0/2)\, \delta(t - u)$, respectively, and $d(t) = a(t)$. The problem is much more general than this example, but the above case is adequate for motivation purposes.

The optimum processor consists of a linear filter that satisfies the equation

$$K_{ar}(t, \sigma) = \int_{T_i}^{t} h_0(t, \tau)\, K_r(\tau, \sigma)\, d\tau, \qquad T_i < \sigma < t. \tag{185}$$

In Section 6.2 we discussed a special case in which $T_i = -\infty$ and the processes were stationary. As part of the solution procedure we found a function $G^+(j\omega)$. We observed that if we passed white noise through a linear system whose transfer function was $G^+(j\omega)$ the output process had a spectrum $S_r(\omega)$. We also observed that in the white noise case, the filter could be realized in what we termed the canonic feedback filter form. The optimum loop filter had the same poles as the linear system whose output spectrum would equal $S_a(\omega)$ if the input were white noise. The only problem was to find the zeros of the optimum loop filter.

For the finite interval it is necessary to solve (185). In Chapter 4 we dealt with similar equations and observed that the conversion of the integral equation to a differential equation with a set of boundary conditions is a useful procedure.

We also observed in several examples that when the message is a scalar Markov process [recall that for a stationary Gaussian process this implies that the covariance had the form $A \exp(-B|t-u|)$] the results were simpler. These observations (plus a great deal of hindsight) lead us to make the following conjectures about an alternate approach to the problem that might be fruitful:

1. Instead of describing the processes of interest in terms of their covariance functions, characterize them in terms of the linear (possibly time-varying) systems that would generate them when driven with white noise.†
2. Instead of describing the linear system that generates the message in terms of a time-varying impulse response, describe it in terms of a differential equation whose solution is the message. The most convenient description will turn out to be a first-order vector differential equation.
3. Instead of specifying the optimum estimate as the output of a linear system which is specified by an integral equation, specify the optimum estimate as the solution to a differential equation whose coefficients are determined by the statistics of the processes. An obvious advantage of this method of specification is that even if we cannot solve the differential equation analytically, we can always solve it easily with an analog or digital computer.

In this section we make these observations more precise and investigate the results.

First, we discuss briefly the state-variable representation of linear, time-varying systems and the generation of random processes. Second, we derive a differential equation which is satisfied by the optimum estimate. Finally, we discuss some applications of the technique.

The original work in this area is due to Kalman and Bucy [23].

6.3.1 Differential Equation Representation of Linear Systems and Random Process Generation‡

In our previous discussions we have characterized linear systems by an impulse response $h(t, u)$ [or simply $h(\tau)$ in the time-invariant case].

† The advantages to be accrued by this characterization were first recognized and exploited by Dolph and Woodbury in 1949 [22].

‡ In this section we develop the background needed to solve the problems of immediate interest. A number of books cover the subject in detail (e.g., Zadeh and DeSoer [24], Gupta [25], Athans and Falb [26], DeRusso, Roy, and Close [27] and Schwartz and Friedland [28]. Our discussion is self-contained, but some results are stated without proof.

Implicit in this description was the assumption that the input was known over the interval $-\infty < t < \infty$. Frequently this method of description is the most convenient. Alternately, we can represent many systems in terms of a differential equation relating its input and output. Indeed, this is the method by which one is usually introduced to linear system theory. The impulse response $h(t, u)$ is just the solution to the differential equation when the input is an impulse at time u.

Three ideas of importance in the differential equation representation are presented in the context of a simple example.

The first idea of importance to us is the idea of initial conditions and state variables in dynamic systems. If we want to find the output over some interval $t_0 \leq t < t_1$, we must know not only the input over this interval but also a certain number of initial conditions that must be adequate to describe how any past inputs ($t < t_0$) affect the output of the system in the interval $t \geq t_0$.

We define the *state* of the system as the minimal amount of information about the effects of past inputs necessary to describe completely the output for $t \geq t_0$. The variables that contain this information are the *state variables*† There must be enough states that every input-output pair can be accounted for. When stated with more mathematical precision, these assumptions imply that, given the state of the system at t_0 and the input from t_0 to t_1, we can find both the *output* and the *state* at t_1. Note that our definition implies that the dynamic systems of interest are deterministic and realizable (future inputs cannot affect the output). If the state can be described by a finite-dimensional vector, we refer to the system as a finite-dimensional dynamic system. In this section we restrict our attention to finite-dimensional systems.

We can illustrate this with a simple example:

Example 1. Consider the *RC* circuit shown in Fig. 6.25. The output voltage $y(t)$ is related to the input voltage $u(t)$ by the differential equation

$$(RC)\,\dot{y}(t) + y(t) = u(t). \tag{186}$$

To find the output $y(t)$ in the interval $t \geq t_0$ we need to know $u(t)$, $t \geq t_0$, and the voltage across the capacitor at t_0. Thus a suitable state variable is $y(t)$.

The second idea is realizing (or simulating) a differential equation by using an analog computer. For our purposes we can visualize an analog computer as a system consisting of integrators, time-varying gains, adders, and nonlinear no-memory devices joined together to produce the desired input-output relation.

For the simple RC circuit example an analog computer realization is shown in Fig. 6.26. The initial condition $y(t_0)$ appears as a bias at the

† Zadeh and DeSoer [24]

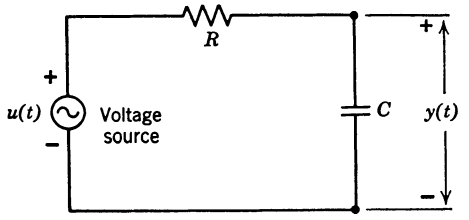


Fig. 6.25 An *RC* circuit.

output of the integrator. This biased integrator output is the state variable of the system.

The third idea is that of random process generation. If $u(t)$ is a random process or $y(t_0)$ is a random variable (or both), then $y(t)$ is a random process. Using the system described by (186), we can generate both nonstationary and stationary processes. As an example of a nonstationary process, let $y(t_0)$ be $N(0, \sigma_0)$, $u(t)$ be zero, and $k = 1/RC$. Then $y(t)$ is a zero-mean Gaussian random process with covariance function

$$K_y(t, u) = \sigma_0^2 e^{-k(t+u-2t_0)}, \qquad t, u \geq t_0. \tag{187}$$

As an example of a stationary process, consider the case in which $u(t)$ is a sample function from a white noise process of spectral height q. If the input starts at $-\infty$ (i.e., $t_0 = -\infty$) and $y(t_0) = 0$, the output is a stationary process with a spectrum

$$S_y(\omega) = \frac{2k\sigma_y^2}{\omega^2 + k^2}, \tag{188}$$

where

$$q = 2\sigma_y^2/k. \tag{189}$$

We now explore these ideas in a more general context. Consider the system described by a differential equation of the form

$$y^{(n)}(t) + p_{n-1}\, y^{(n-1)}(t) + \cdots + p_0\, y(t) = b_0\, u(t), \tag{190}$$

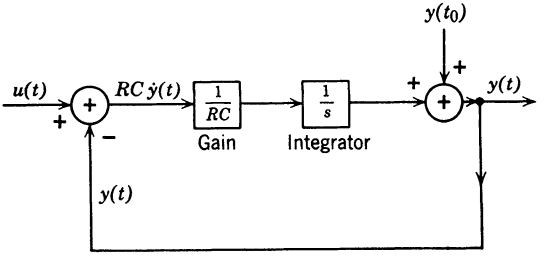


Fig. 6.26 An analog computer realization.

where $y^{(n)}(t)$ denotes the nth derivative of $y(t)$. Recall that to specify the solution to an nth-order equation we need the values of $y(t), \ldots, y^{(n-1)}(t)$ at t_0. This observation will be the key to finding the state representation for this system. The first step in finding an analog computer realization is to generate the terms on the left-hand side of the equation. This is shown in Fig. 6.27*a*. The next step is to interconnect these various quantities so that the differential equation is satisfied. The differential equation specifies the inputs to the summing point and gives the block diagram shown in Fig. 6.27*b*. Finally, we include the initial conditions by allowing

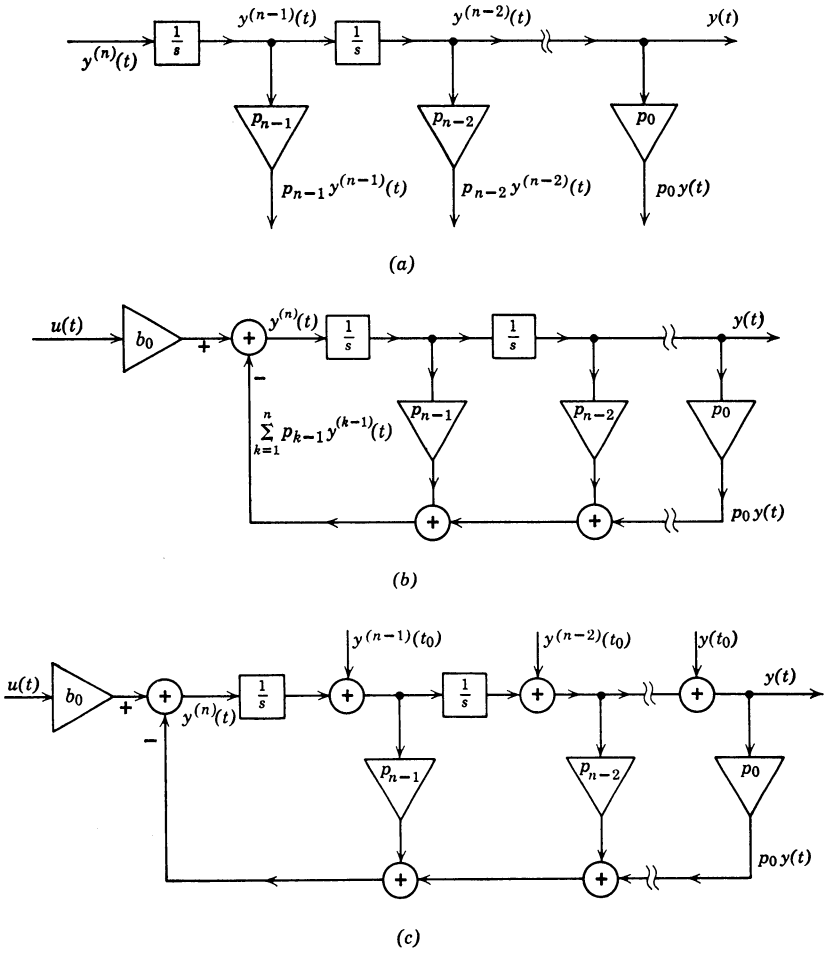


Fig. 6.27 Analog computer realization.

for a bias on the integrator outputs and obtain the realization shown in Fig. 6.27*c*. The state variables are the biased integrator outputs.

It is frequently easier to work with a first-order vector differential equation than an nth-order scalar equation. For (190) the transformation is straightforward. Let

$$\begin{aligned} x_1(t) &= y(t), \\ x_2(t) &= \dot{y}(t) = \dot{x}_1(t), \\ &\vdots \\ x_n(t) &= y^{(n-1)}(t) = \dot{x}_{n-1}(t). \end{aligned} \tag{191}$$

$$\dot{x}_n(t) = y^{(n)}(t) = -\sum_{k=1}^{n} p_{k-1} y^{(k-1)}(t) + b_0 u(t)$$

$$= -\sum_{k=1}^{n} p_{k-1} x_k(t) + b_0 u(t). \tag{191}$$

Denoting the set of $x_i(t)$ by a column matrix, we see that the following first-order n-dimensional vector equation is equivalent to the nth-order scalar equation.

$$\frac{d\mathbf{x}(t)}{dt} \triangleq \dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}u(t), \tag{192}$$

where

$$\mathbf{F} = \left[\begin{array}{c|ccccc} 0 & 1 & & & & \\ 0 & & 1 & & 0 & \\ 0 & & & 1 & & \\ \vdots & & 0 & & \ddots & \\ 0 & & & & & 1 \\ \hline -p_0 & -p_1 & -p_2 & -p_3 & \cdots & -p_{n-1} \end{array}\right] \tag{193}$$

and

$$\mathbf{G} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_0 \end{bmatrix}. \tag{194}$$

The vector $\mathbf{x}(t)$ is called the *state vector* for this linear system and (192) is called the *state equation* of the system. Note that the state vector $\mathbf{x}(t)$ we selected is not the only choice. Any nonsingular linear transformation of $\mathbf{x}(t)$ gives another state vector. The output $y(t)$ is related to the state vector by the equation

$$y(t) = \mathbf{C}\,\mathbf{x}(t), \tag{195}$$

where $\mathbf{C}$ is a $1 \times n$ matrix

$$\mathbf{C} = [1 \,\vdots\, 0 \,\vdots\, 0 \,\vdots\, 0 \cdots 0]. \tag{196}$$

Equation (195) is called the *output equation* of the system. The two equations (192) and (195) completely characterize the system.

Just as in the first example we can generate both nonstationary and stationary random processes using the system described by (192) and (195). For stationary processes it is clear (190) that we can generate any process with a rational spectrum in the form of

$$S_y(\omega) = \frac{k}{d_{2n}\omega^{2n} + d_{2n-2}\omega^{2n-2} + \cdots + d_0} \tag{197}$$

by letting $u(t)$ be a white noise process and $t_0 = -\infty$. In this case the state vector $\mathbf{x}(t)$ is a sample function from a vector random process and $y(t)$ is one component of this process.

The next more general differential equation is

$$y^{(n)}(t) + p_{n-1}\, y^{(n-1)}(t) + \cdots + p_0\, y(t) = b_{n-1}\, u^{(n-1)}(t) + \cdots + b_0 u(t). \tag{198}$$

The first step is to find an analog computer-type realization that corresponds to this differential equation. We illustrate one possible technique by looking at a simple example.

Example 2A. Consider the case in which $n = 2$ and the initial conditions are zero. Then (198) is

$$\ddot{y}(t) + p_1\, \dot{y}(t) + p_0\, y(t) = b_1\, \dot{u}(t) + b_0\, u(t). \tag{199}$$

Our first observation is that we want to avoid actually differentiating $u(t)$ because in many cases of interest it is a white noise process. Comparing the order of the highest derivatives on the two sides of (199), we see that this is possible. An easy approach is to assume that $\dot{u}(t)$ exists as part of the input to the first integrator in Fig. 6.28 and examine the consequences. To do this we rearrange terms as shown in (200):

$$[\ddot{y}(t) - b_1\, \dot{u}(t)] + p_1\, \dot{y}(t) + p_0\, y(t) = b_0\, u(t). \tag{200}$$

The result is shown in Fig. 6.28. Defining the state variables as the integrator outputs, we obtain

$$x_1(t) = y(t) \tag{201a}$$

and

$$x_2(t) = \dot{y}(t) - b_1\, u(t). \tag{201b}$$

Using (200) and (201), we have

$$\dot{x}_1(t) = x_2(t) + b_1\, u(t) \tag{202a}$$

$$\begin{aligned} \dot{x}_2(t) &= -p_0\, x_1(t) - p_1\, (x_2(t) + b_1\, u(t)) + b_0\, u(t) \\ &= -p_0\, x_1(t) - p_1\, x_2(t) + (b_0 - b_1 p_1)\, u(t). \end{aligned} \tag{202b}$$

We can write (202) as a vector state equation by defining

$$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ -p_0 & -p_1 \end{bmatrix} \tag{203a}$$

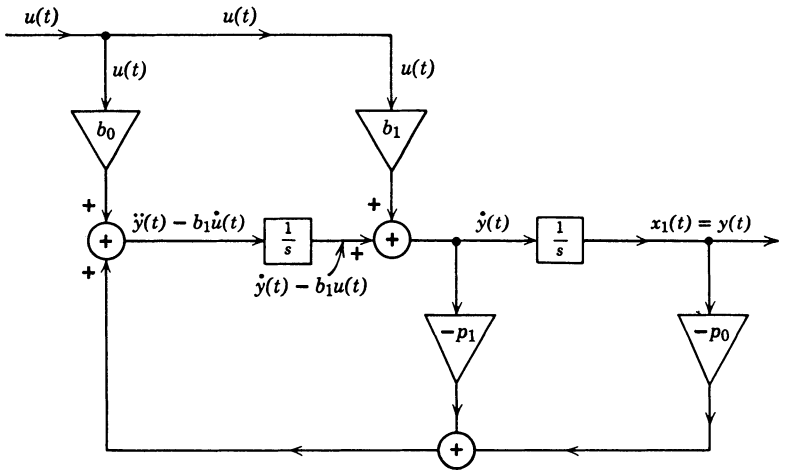


Fig. 6.28 Analog realization.

and

$$\mathbf{G} = \begin{bmatrix} b_1 \\ b_0 - p_1 b_1 \end{bmatrix}. \tag{203b}$$

Then

$$\dot{\mathbf{x}}(t) = \mathbf{F}\,\mathbf{x}(t) + \mathbf{G}\,u(t). \tag{204a}$$

The output equation is

$$y(t) = [1 \;\vdots\; 0]\mathbf{x}(t) \triangleq \mathbf{C}\,\mathbf{x}(t). \tag{204b}$$

Equations 204*a* and 204*b* plus the initial condition $\mathbf{x}(t_0) = \mathbf{0}$ characterize the system.

It is straightforward to extend this particular technique to the *n*th order (see Problem 6.3.1). We refer to it as canonical realization No. 1. Our choice of state variables was somewhat arbitrary. To demonstrate this, we reconsider Example 2*A* and develop a different state representation.

Example 2B. Once again

$$\ddot{y}(t) + p_1\,\dot{y}(t) + p_0\,y(t) = b_1\,\dot{u}(t) + b_0\,u(t). \tag{205}$$

As a first step we draw the two integrators and the two paths caused by b_1 and b_0. This partial system is shown in Fig. 6.29*a*. We now want to introduce feedback paths and identify state variables in such a way that the elements in **F** and **G** will be one of the coefficients in the original differential equation, unity, or zero. Looking at Fig. 6.29*a*, we see that an easy way to do this is to feed back a weighted version of $x_1(t)$ $(= y(t))$ into each summing point as shown in Fig. 6.29*b*. The equations for the state variables are

$$x_1(t) = y(t), \tag{206}$$

$$\dot{x}_1(t) = x_2(t) - p_1\,y(t) + b_1\,u(t), \tag{207}$$

$$\dot{x}_2(t) = -p_0\,y(t) + b_0\,u(t). \tag{208}$$

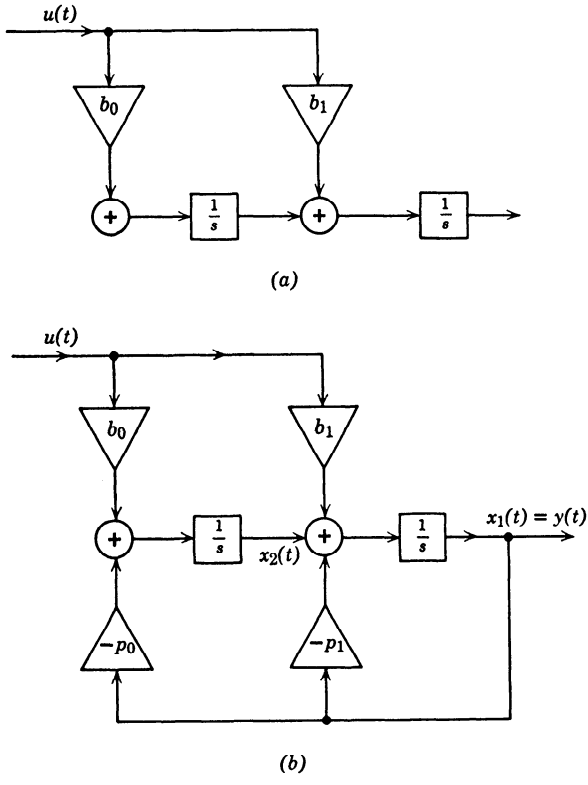


Fig. 6.29 Analog realization of (205).

The **F** matrix is

$$\mathbf{F} = \begin{bmatrix} -p_1 & +1 \\ -p_0 & 0 \end{bmatrix} \tag{209}$$

and the **G** matrix is

$$\mathbf{G} = \begin{bmatrix} b_1 \\ b_0 \end{bmatrix}. \tag{210}$$

We see that the system has the desired property.

The extension to the original nth-order differential equation is straightforward. The resulting realization is shown in Fig. 6.30. The equations for the state variables are

$$\begin{aligned} x_1(t) &= y(t), \\ x_2(t) &= \dot{x}_1(t) + p_{n-1}y(t) - b_{n-1}u(t), \\ &\vdots \\ x_n(t) &= \dot{x}_{n-1}(t) + p_1 y(t) - b_1 u(t), \\ \dot{x}_n(t) &= -p_0 y(t) + b_0 u(t). \end{aligned} \tag{211}$$

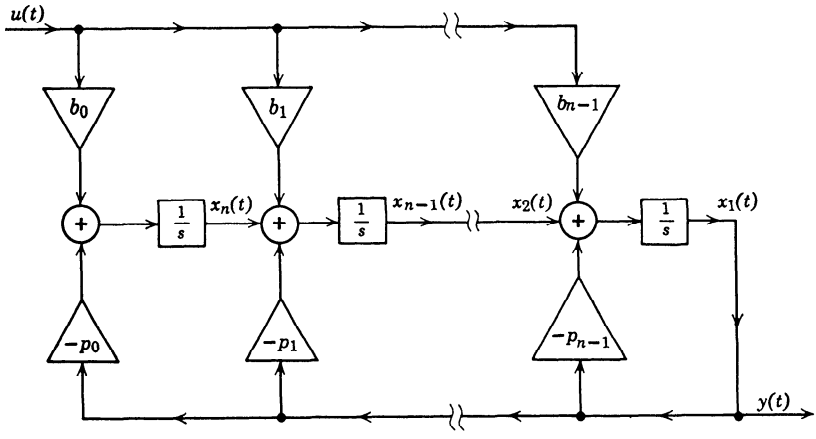


Fig. 6.30 Canonic realization No. 2: state variables.

The matrix for the vector differential equation is

$$\mathbf{F} = \left[\begin{array}{c|ccc} -p_{n-1} & 1 & 0 & \\ -p_{n-2} & 0 & 1 & 0 \\ & & & 1 \\ \vdots & & & \ddots \\ & & 0 & \\ -p_1 & & & 1 \\ \hline -p_0 & 0 & \cdots & 0 \end{array}\right] \tag{212}$$

and

$$\mathbf{G} = \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_0 \end{bmatrix}. \tag{213}$$

We refer to this realization as canonical realization No. 2.

There is still a third useful realization to consider. The transfer function corresponding to (198) is

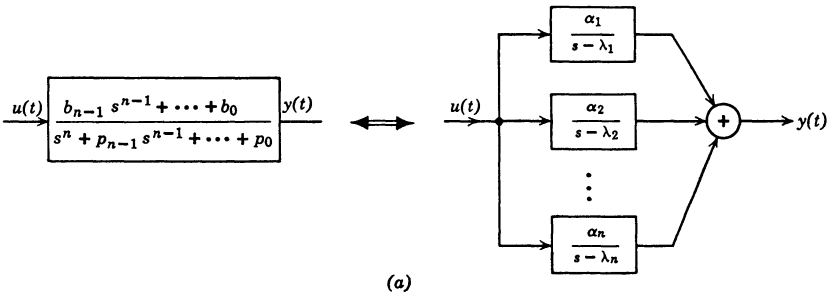
$$\frac{Y(s)}{X(s)} = \frac{b_{n-1}s^{n-1} + \cdots + b_0}{s^n + p_{n-1}s^{n-1} + \cdots + p_0} \triangleq H(s). \tag{214}$$

We can expand this equation in a partial fraction expansion

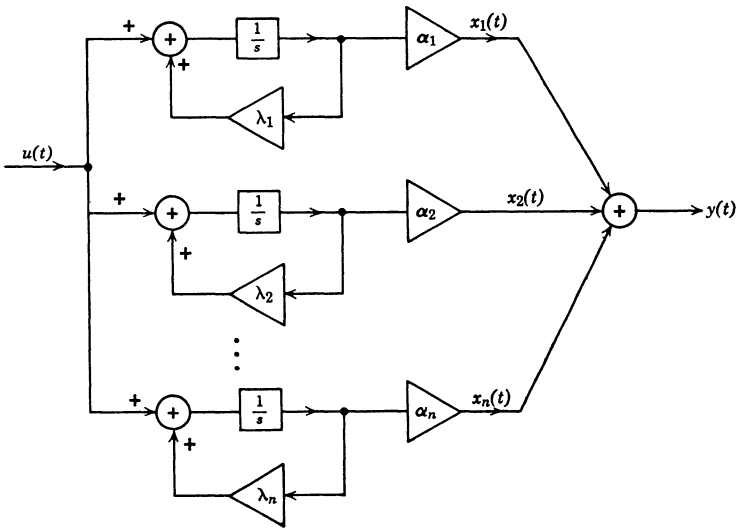
$$H(s) = \sum_{i=1}^{n} \frac{\alpha_i}{s - \lambda_i}, \tag{215}$$

where the λ_i are the roots of the denominator that are assumed to be distinct and the α_i are the corresponding residues. The system is shown in transform notation in Fig. 6.31*a*. Clearly, we can identify each subsystem output as a state variable and realize the over-all system as shown in Fig. 6.31*b*. The **F** matrix is diagonal.

$$\mathbf{F} = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & 0 \\ & & \lambda_3 & \\ & 0 & & \ddots \\ & & & & \lambda_n \end{bmatrix} \tag{216}$$



(*a*)



(*b*)

Fig. 6.31 Canonic realization No. 3: (*a*) transfer function; (*b*) analog computer realization.

and the elements in the **G** matrix are the residues

$$\mathbf{G} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}. \tag{217}$$

Now the original output $y(t)$ is the sum of the state variables

$$y(t) = \sum_{i=1}^{n} x_i(t) = \mathbf{1}^T\mathbf{x}(t), \tag{218a}$$

where

$$\mathbf{1}^T \triangleq [1 \mid 1 \cdots 1]. \tag{218b}$$

We refer to this realization as canonical realization No. 3. (The realization for repeated roots is derived in Problem 6.3.2.)

Canonical realization No. 3 requires a partial fraction expansion to find **F** and **G**. Observe that the state equation consists of *n uncoupled* first-order scalar equations

$$\dot{x}_i = \lambda_i x_i(t) + \alpha_i u(t), \qquad i = 1, 2, \ldots, n. \tag{219}$$

The solution of this set is appreciably simpler than the solution of the vector equation. On the other hand, finding the partial fraction expansion may require some calculation whereas canonical realizations No. 1 and No. 2 can be obtained by inspection.

We have now developed three different methods for realizing a system described by an nth-order constant coefficient differential equation. In each case the state vector was different. The **F** matrices were different, but it is easy to verify that they all have the same eigenvalues. It is worthwhile to emphasize that even though we have labeled these realizations as canonical some other realization may be more desirable in a particular problem. Any nonsingular linear transformation of a state vector leads to a new state representation.

We now have the capability of generating *any* stationary random process with a rational spectrum and finite variance by exciting any of the three realizations with white noise. In addition we can generate a wide class of nonstationary processes.

Up to this point we have seen how to represent linear time-invariant systems in terms of a state-variable representation and the associated vector-differential equation. We saw that this could correspond to a physical realization in the form of an analog computer, and we learned how we could generate a large class of random processes.

The next step is to extend our discussion to include time-varying systems and multiple input-multiple output systems.

For time-varying systems we consider the vector equations

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,u(t), \tag{220a}$$

$$y(t) = \mathbf{C}(t)\,\mathbf{x}(t), \tag{220b}$$

as the basic representation.† The matrices $\mathbf{F}(t)$ and $\mathbf{G}(t)$ may be functions of time. By using a white noise input

$$E[u(t)\,u(\tau)] = q\,\delta(t - \tau), \tag{221}$$

we have the ability to generate some nonstationary random processes. It is worthwhile to observe that a nonstationary process can result even when $\mathbf{F}$ and $\mathbf{G}$ are constant and $\mathbf{x}(t_0)$ is deterministic. The Wiener process, defined on p. 195 of Chapter 3, is a good example.

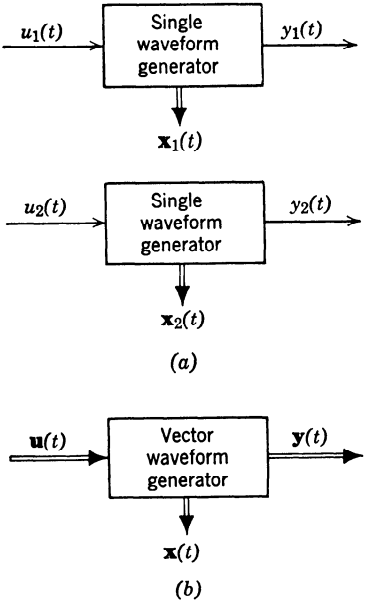


Fig. 6.32 Generation of two messages.

† The canonic realizations in Figs. 6.28 and 6.30 may still be used. It is important to observe that they do not correspond to the same nth-order differential equation as in the time-invariant case. See Problem 6.3.14.

Example 3. Here $\mathbf{F}(t) = 0$, $\mathbf{G}(t) = \sigma$, $\mathbf{C}(t) = 1$, and (220) becomes

$$\frac{dx(t)}{dt} = \sigma\, u(t). \tag{222}$$

Assuming that $x(0) = 0$, this gives the Wiener process.

Other specific examples of time-varying systems are discussed in later sections and in the problems.

The motivation for studying multiple input–multiple output systems follows directly from our discussions in Chapters 3, 4, and 5. Consider the simple system in Fig. 6.32 in which we generate two outputs $y_1(t)$ and $y_2(t)$. We assume that the state representation of system 1 is

$$\dot{\mathbf{x}}_1(t) = \mathbf{F}_1(t)\,\mathbf{x}_1(t) + \mathbf{G}_1(t)\,u_1(t), \tag{223}$$

$$y_1(t) = \mathbf{C}_1(t)\,\mathbf{x}_1(t), \tag{224}$$

where $\mathbf{x}_1(t)$ is an n-dimensional state vector. Similarly, the state representation of system 2 is

$$\dot{\mathbf{x}}_2(t) = \mathbf{F}_2(t)\,\mathbf{x}_2(t) + \mathbf{G}_2(t)\,u_2(t), \tag{225}$$

$$y_2(t) = \mathbf{C}_2(t)\,\mathbf{x}_2(t), \tag{226}$$

where $\mathbf{x}_2(t)$ is an m-dimensional state vector. A more convenient way to describe these two systems is as a single vector system with an $(n + m)$-dimensional state vector (Fig. 6.32*b*).

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \hline \mathbf{x}_2(t) \end{bmatrix}, \tag{227}$$

$$\mathbf{F}(t) = \left[\begin{array}{c|c} \mathbf{F}_1(t) & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{F}_2(t) \end{array}\right], \tag{228}$$

$$\mathbf{G}(t) = \left[\begin{array}{c|c} \mathbf{G}_1(t) & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{G}_2(t) \end{array}\right], \tag{229}$$

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ \hline u_2(t) \end{bmatrix}, \tag{230}$$

$$\mathbf{C}(t) = \left[\begin{array}{c|c} \mathbf{C}_1(t) & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{C}_2(t) \end{array}\right], \tag{231}$$

and

$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ \hline y_2(t) \end{bmatrix}. \tag{232}$$

The resulting differential equations are

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{u}(t), \tag{233}$$

$$\mathbf{y}(t) = \mathbf{C}(t)\,\mathbf{x}(t). \tag{234}$$

The driving function is a vector. For the message generation problem we assume that the driving function is a white process with a matrix covariance function

$$E[\mathbf{u}(t)\,\mathbf{u}^T(\tau)] \triangleq \mathbf{Q}\,\delta(t - \tau), \tag{235}$$

where $\mathbf{Q}$ is a nonnegative definite matrix. The block diagram of the generation process is shown in Fig. 6.33.

Observe that in general the initial conditions may be random variables. Then, to specify the second-moment characteristics we must know the covariance at the initial time

$$\mathbf{K}_x(t_0, t_0) \triangleq E[\mathbf{x}(t_0)\,\mathbf{x}^T(t_0)] \tag{236}$$

and the mean value $E[\mathbf{x}(t_0)]$. We can also generate coupled processes by replacing the $\mathbf{0}$ matrices in (228), (229), or (231) with nonzero matrices.

The next step in our discussion is to consider the solution to (233). We begin our discussion with the *homogeneous time-invariant* case. Then (233) reduces to

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t), \tag{237}$$

with initial condition $\mathbf{x}(t_0)$. If $\mathbf{x}(t)$ and $\mathbf{F}$ are scalars, the solution is familiar,

$$x(t) = e^{F(t-t_0)}x(t_0). \tag{238}$$

For the vector case we can show that (e.g. [27], [28], [29], or [30])

$$\mathbf{x}(t) = e^{\mathbf{F}(t-t_0)}\,\mathbf{x}(t_0), \tag{239}$$

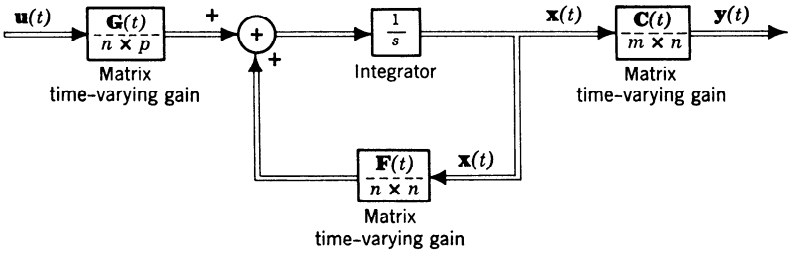


Fig. 6.33 Message generation process.

where $e^{\mathbf{F}t}$ is defined by the infinite series

$$e^{\mathbf{F}t} \triangleq \mathbf{I} + \mathbf{F}t + \frac{\mathbf{F}^2t^2}{2!} + \cdots. \tag{240}$$

The function $e^{\mathbf{F}(t-t_0)}$ is denoted by $\boldsymbol{\phi}(t - t_0) \triangleq \boldsymbol{\phi}(\tau)$. The function $\boldsymbol{\phi}(t - t_0)$ is called the *state transition matrix* of the system. Two properties can easily be verified for the time-invariant case.

Property 11.† The state transition matrix satisfies the equation

$$\frac{d\boldsymbol{\phi}(t - t_0)}{dt} = \mathbf{F}\boldsymbol{\phi}(t - t_0) \tag{241}$$

or

$$\frac{d\boldsymbol{\phi}(\tau)}{d\tau} = \mathbf{F}\boldsymbol{\phi}(\tau). \tag{242}$$

[Use (240) and its derivative on both sides of (239).]

Property 12. The initial condition

$$\boldsymbol{\phi}(t_0 - t_0) = \boldsymbol{\phi}(0) = \mathbf{I} \tag{243}$$

follows directly from (239). The homogeneous solution can be rewritten in terms of $\boldsymbol{\phi}(t - t_0)$:

$$\mathbf{x}(t) = \boldsymbol{\phi}(t - t_0)\,\mathbf{x}(t_0). \tag{244}$$

The solution to (242) is easily obtained by using conventional Laplace transform techniques. Transforming (242), we have

$$s\,\boldsymbol{\Phi}(s) - \mathbf{I} = \mathbf{F}\,\boldsymbol{\Phi}(s), \tag{245}$$

where the identity matrix arises from the initial condition in (243). Rearranging terms, we have

$$[s\mathbf{I} - \mathbf{F}]\,\boldsymbol{\Phi}(s) = \mathbf{I} \tag{246}$$

or

$$\boldsymbol{\Phi}(s) = (s\mathbf{I} - \mathbf{F})^{-1}. \tag{247}$$

The state transition matrix is

$$\boldsymbol{\phi}(\tau) = \mathcal{L}^{-1}[\boldsymbol{\Phi}(s)] = \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{F})^{-1}]. \tag{248}$$

A simple example illustrates the technique.

Example 4. Consider the system described by (206–210). The transform of the transition matrix is,

$$\boldsymbol{\Phi}(s) = \left[s\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} - \begin{bmatrix}-p_1 & 1\\ -p_0 & 0\end{bmatrix}\right]^{-1}, \tag{249}$$

† Because we have to refer back to the properties at the beginning of the chapter we use a consecutive numbering system to avoid confusion.

$$\mathbf{\Phi}(s) = \begin{bmatrix} s + p_1 & -1 \\ p_0 & s \end{bmatrix}^{-1}, \tag{250}$$

$$\mathbf{\Phi}(s) = \frac{1}{s^2 + p_1 s + p_0} \begin{bmatrix} s & 1 \\ -p_0 & s + p_1 \end{bmatrix}. \tag{251}$$

To find $\boldsymbol{\phi}(\tau)$ we take the inverse transform. For simplicity we let $p_1 = 3$ and $p_0 = 2$. Then

$$\boldsymbol{\phi}(\tau) = \left[\begin{array}{c|c} 2e^{-2\tau} - e^{-\tau} & e^{-\tau} - e^{-2\tau} \\ \hline 2[e^{-2\tau} - e^{-\tau}] & 2e^{-\tau} - e^{-2\tau} \end{array}\right]. \tag{252}$$

It is important to observe that the complex natural frequencies involved in the solution are determined by the denominator of $\mathbf{\Phi}(s)$. This is just the determinant of the matrix $s\mathbf{I} - \mathbf{F}$. Therefore these frequencies are just the roots of the equation

$$\det [s\mathbf{I} - \mathbf{F}] = 0. \tag{253}$$

For the time-varying case the basic concept of a state-transition matrix is still valid, but some of the above properties no longer hold. From the scalar case we know that $\boldsymbol{\phi}(t, t_0)$ will be a function of two variables instead of just the difference between t and t_0.

Definition. The state transition matrix is defined to be a function of two variables $\boldsymbol{\phi}(t, t_0)$ which satisfies the differential equation

$$\dot{\boldsymbol{\phi}}(t, t_0) = \mathbf{F}(t)\boldsymbol{\phi}(t, t_0) \tag{254a}$$

with initial condition $\boldsymbol{\phi}(t_0, t_0) = \mathbf{I}$. The solution at any time is

$$\mathbf{x}(t) = \boldsymbol{\phi}(t, t_0)\,\mathbf{x}(t_0). \tag{254b}$$

An analytic solution is normally difficult to obtain. Fortunately, in most of the cases in which we use the transition matrix an analytic solution is not necessary. Usually, we need only to know that it exists and that it has certain properties. In the cases in which it actually needs evaluation, we shall do it numerically.

Two properties follow easily:

$$\boldsymbol{\phi}(t_2, t_0) = \boldsymbol{\phi}(t_2, t_1)\boldsymbol{\phi}(t_1, t_0), \qquad \text{for all } t_0, t_1, t_2 \tag{255a}$$

and

$$\boldsymbol{\phi}^{-1}(t_1, t_0) = \boldsymbol{\phi}(t_0, t_1). \tag{255b}$$

For the nonhomogeneous case the equation is

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{u}(t). \tag{256}$$

The solution contains a homogeneous part and a particular part:

$$\mathbf{x}(t) = \boldsymbol{\phi}(t, t_0)\,\mathbf{x}(t_0) + \int_{t_0}^{t} \boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau)\,\mathbf{u}(\tau)\,d\tau. \tag{257}$$

(Substitute (257) into (256) to verify that it is the solution.) The output $\mathbf{y}(t)$ is

$$\mathbf{y}(t) = \mathbf{C}(t)\,\mathbf{x}(t). \tag{258}$$

In our work in Chapters 4 and 5 and Section 6.1 we characterized time-varying linear systems by their impulse response $\mathbf{h}(t, \tau)$. This characterization assumes that the input is known from $-\infty$ to t. Thus

$$\mathbf{y}(t) = \int_{-\infty}^{t} \mathbf{h}(t, \tau)\,\mathbf{u}(\tau)\,d\tau. \tag{259}$$

For most cases of interest the effect of the initial condition $\mathbf{x}(-\infty)$ will disappear in (257). Therefore, we may set them equal to zero and obtain,

$$\mathbf{y}(t) = \mathbf{C}(t) \int_{-\infty}^{t} \boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau)\,\mathbf{u}(\tau)\,d\tau. \tag{260}$$

Comparing (259) and (260), we have

$$\begin{aligned} \mathbf{h}(t, \tau) &= \mathbf{C}(t)\boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau), && t \geq \tau, \\ &\mathbf{0}, && \text{elsewhere.} \end{aligned} \tag{261}$$

It is worthwhile to observe that the three matrices on the right will depend on the state representation that we choose for the system, but the matrix impulse response is unique. As pointed out earlier, the system is realizable. This is reflected by the $\mathbf{0}$ in (261).

For the time-invariant case

$$\mathbf{Y}(s) = \mathbf{H}(s)\,\mathbf{U}(s), \tag{262}$$

and

$$\mathbf{H}(s) = \mathbf{C}\,\boldsymbol{\Phi}(s)\mathbf{G}. \tag{263}$$

Equation 262 assumes that the input has a Laplace transform. For a stationary random process we would use the integrated transform (Section 3.6).

Most of our discussion up to this point has been valid for an arbitrary driving function $\mathbf{u}(t)$. We now derive some statistical properties of vector processes $\mathbf{x}(t)$ and $\mathbf{y}(t)$ for the specific case in which $\mathbf{u}(t)$ is a sample function of a vector white noise process.

$$E[\mathbf{u}(t)\,\mathbf{u}^T(\tau)] = \mathbf{Q}\,\delta(t - \tau). \tag{264}$$

Property 13. The cross correlation between the state vector $\mathbf{x}(t)$ of a system driven by a zero-mean white noise $\mathbf{u}(t)$ and the input $\mathbf{u}(\tau)$ is

$$\mathbf{K}_{\mathbf{xu}}(t, \tau) \triangleq E[\mathbf{x}(t)\,\mathbf{u}^T(\tau)]. \tag{265}$$

It is a discontinuous function that equals

$$\mathbf{K}_{\mathbf{xu}}(t, \tau) = \begin{cases} \mathbf{0}, & \tau > t, \\ \tfrac{1}{2}\mathbf{G}(t)\mathbf{Q}, & \tau = t, \\ \boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau)\mathbf{Q}, & t_0 < \tau < t. \end{cases} \tag{266}$$

Proof. Substituting (257) into the definition in (265), we have

$$\mathbf{K}_{\mathbf{xu}}(t, \tau) = E\left\{\left[\boldsymbol{\phi}(t, t_0)\,\mathbf{x}(t_0) + \int_{t_0}^{t} \boldsymbol{\phi}(t, \alpha)\,\mathbf{G}(\alpha)\,\mathbf{u}(\alpha)\,d\alpha\right]\mathbf{u}^T(\tau)\right\}. \quad (267)$$

Bringing the expectation inside the integral and assuming that the initial state $\mathbf{x}(t_0)$ is independent of $\mathbf{u}(\tau)$ for $\tau > t_0$, we have

$$\begin{aligned}\mathbf{K}_{\mathbf{xu}}(t, \tau) &= \int_{t_0}^{t} \boldsymbol{\phi}(t, \alpha)\,\mathbf{G}(\alpha)\,E[\mathbf{u}(\alpha)\,\mathbf{u}^T(\tau)]\,d\alpha \\ &= \int_{t_0}^{t} \boldsymbol{\phi}(t, \alpha)\,\mathbf{G}(\alpha)\,\mathbf{Q}\,\delta(\alpha - \tau)\,d\alpha. \end{aligned} \quad (268)$$

If $\tau > t$, this expression is zero. If $\tau = t$ and we assume that the delta function is symmetric because it is the limit of a covariance function, we pick up only one half the area at the right end point. Thus

$$\mathbf{K}_{\mathbf{xu}}(t, t) = \tfrac{1}{2}\boldsymbol{\phi}(t, t)\,\mathbf{G}(t)\mathbf{Q}. \quad (269)$$

Using the result following (254*a*), we obtain the second line in (266).

If $\tau < t$, we have

$$\mathbf{K}_{\mathbf{xu}}(t, \tau) = \boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau)\mathbf{Q}, \qquad \tau < t \quad (270a)$$

which is the third line in (266). A special case of (270*a*) that we shall use later is obtained by letting τ approach t from below.

$$\lim_{\tau \to t^-} \mathbf{K}_{\mathbf{xu}}(t, \tau) = \mathbf{G}(t)\mathbf{Q}. \quad (270b)$$

The cross correlation between the output vector $\mathbf{y}(t)$ and $\mathbf{u}(\tau)$ follows easily.

$$\mathbf{K}_{\mathbf{yu}}(t, \tau) = \mathbf{C}(t)\,\mathbf{K}_{\mathbf{xu}}(t, \tau). \quad (271)$$

Property 14. The variance matrix of the state vector $\mathbf{x}(t)$ of a system

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{u}(t) \quad (272)$$

satisfies the differential equation

$$\dot{\boldsymbol{\Lambda}}_{\mathbf{x}}(t) = \mathbf{F}(t)\,\boldsymbol{\Lambda}_{\mathbf{x}}(t) + \boldsymbol{\Lambda}_{\mathbf{x}}(t)\,\mathbf{F}^T(t) + \mathbf{G}(t)\,\mathbf{Q}\,\mathbf{G}^T(t), \quad (273)$$

with the initial condition

$$\boldsymbol{\Lambda}_{\mathbf{x}}(t_0) = E[\mathbf{x}(t_0)\,\mathbf{x}^T(t_0)]. \quad (274)$$

[Observe that $\boldsymbol{\Lambda}_{\mathbf{x}}(t) = \mathbf{K}_{\mathbf{x}}(t, t)$.]

Proof.

$$\boldsymbol{\Lambda}_{\mathbf{x}}(t) \triangleq E[\mathbf{x}(t)\,\mathbf{x}^T(t)]. \quad (275)$$

Differentiating, we have

$$\frac{d\Lambda_{\mathbf{x}}(t)}{dt} = E\left[\frac{d\mathbf{x}(t)}{dt}\,\mathbf{x}^T(t)\right] + E\left[\mathbf{x}(t)\,\frac{d\mathbf{x}^T(t)}{dt}\right]. \tag{276}$$

The second term is just the transpose of the first. [Observe that $\mathbf{x}(t)$ is not mean-square differentiable: therefore, we will have to be careful when dealing with (276).]

Substituting (272) into the first term in (276) gives

$$E\left[\frac{d\mathbf{x}(t)}{dt}\,\mathbf{x}^T(t)\right] = E\{[\mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{u}(t)]\,\mathbf{x}^T(t)\}. \tag{277}$$

Using Property 13 on the second term in (277), we have

$$E\left[\frac{d\mathbf{x}(t)}{dt}\,\mathbf{x}^T(t)\right] = \mathbf{F}(t)\,\Lambda_{\mathbf{x}}(t) + \tfrac{1}{2}\mathbf{G}(t)\,\mathbf{Q}\,\mathbf{G}^T(t). \tag{278}$$

Using (278) and its transpose in (276) gives

$$\dot{\Lambda}_{\mathbf{x}}(t) = \mathbf{F}(t)\,\Lambda_{\mathbf{x}}(t) + \Lambda_{\mathbf{x}}(t)\,\mathbf{F}^T(t) + \mathbf{G}(t)\,\mathbf{Q}\,\mathbf{G}^T(t), \tag{279}$$

which is the desired result.

We now have developed the following ideas:

1. State variables of a linear dynamic system;
2. Analog computer realizations;
3. First-order vector differential equations and state-transition matrices;
4. Random process generation.

The next step is to apply these ideas to the linear estimation problem.

Observation Model. In this section we recast the linear modulation problem described in the beginning of the chapter into state-variable terminology. The basic linear modulation problem was illustrated in Fig. 6.1. A state-variable formulation for a simpler special case is given in Fig. 6.34. The message $a(t)$ is generated by passing $y(t)$ through a linear system, as discussed in the preceding section. Thus

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,u(t) \tag{280}$$

and

$$a(t) \triangleq x_1(t). \tag{281}$$

For simplicity in the explanation we have assumed that the message of interest is the first component of the state vector. The message is then modulated by multiplying by the carrier $c(t)$. In DSB-AM, $c(t)$ would be a sine wave. We include the carrier in the linear system by defining

$$\mathbf{C}(t) = [c(t) \,\vdots\, 0 \,\vdots\, 0 \quad \cdots \quad 0]. \tag{282}$$

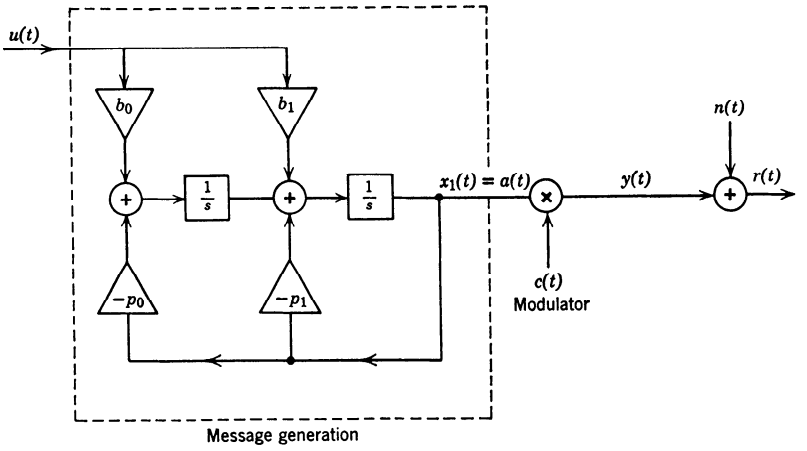


Fig. 6.34 A simple case of linear modulation in the state-variable formulation.

Then

$$y(t) = \mathbf{C}(t)\,\mathbf{x}(t). \tag{283}$$

We frequently refer to $\mathbf{C}(t)$ as the *modulation matrix.* (In the control literature it is called the *observation matrix.*) The waveform $y(t)$ is transmitted over an additive white noise channel. Thus

$$\begin{aligned} r(t) &= y(t) + w(t), \qquad T_i \le t \le T_f \\ &= \mathbf{C}(t)\,\mathbf{x}(t) + w(t), \qquad T_i \le t \le T_f, \end{aligned} \tag{284}$$

where

$$E[w(t)\,w(\tau)] = \frac{N_0}{2}\,\delta(t - \tau). \tag{285}$$

This particular model is too restrictive; therefore we generalize it in several different ways. Two of the modifications are fundamental and we explain them now. The others are deferred until Section 6.3.4 to avoid excess notation at this point.

Modification No. 1: Colored Noise. In this case there is a colored noise component $n_c(t)$ in addition to the white noise. We assume that the colored noise can be generated by driving a finite-dimensional dynamic system with white noise.

$$r(t) = \mathbf{C}_M(t)\,\mathbf{x}_M(t) + n_c(t) + w(t). \tag{286}$$

The subscript M denotes message. We can write (286) in the form

$$r(t) = \mathbf{C}(t)\,\mathbf{x}(t) + w(t) \tag{287}$$

by augmenting the message state vector to include the colored noise process. The new vector process $\mathbf{x}(t)$ consists of two parts. One is the vector process $\mathbf{x}_M(t)$ corresponding to the state variables of the system used to generate the message process. The second is the vector process $\mathbf{x}_N(t)$ corresponding to the state variables of the system used to generate the colored noise process. Thus

$$\mathbf{x}(t) \triangleq \begin{bmatrix} \mathbf{x}_M(t) \\ \mathbf{x}_N(t) \end{bmatrix}. \tag{288}$$

If $\mathbf{x}_M(t)$ is n_1-dimensional and $\mathbf{x}_N(t)$ is n_2-dimensional, then $\mathbf{x}(t)$ has $(n_1 + n_2)$ dimensions. The modulation matrix is

$$\mathbf{C}(t) = [\mathbf{C}_M(t) \,\vdots\, \mathbf{C}_N(t)]; \tag{289}$$

$\mathbf{C}_M(t)$ is defined in (286) and $\mathbf{C}_N(t)$ is chosen so that

$$n_c(t) = \mathbf{C}_N(t)\,\mathbf{x}_N(t). \tag{290}$$

With these definitions, we obtain (287). A simple example is appropriate at this point.

Example. Let

$$r(t) = \sqrt{2P}\,a(t)\sin \omega_c t + n_c(t) + w(t), \qquad -\infty < t, \tag{291}$$

where

$$S_a(\omega) = \frac{2k_aP_a}{\omega^2 + k_a{}^2}, \tag{292}$$

$$S_{n_c}(\omega) = \frac{2k_nP_n}{\omega^2 + k_n{}^2}, \tag{293}$$

and $n_c(t)$, $a(t)$, and $w(t)$ are uncorrelated. To obtain a state representation we let $t_0 = -\infty$ and assume that $a(-\infty) = n_c(-\infty) = 0$. We define the state vector $\mathbf{x}(t)$ as

$$\mathbf{x}(t) = \begin{bmatrix} a(t) \\ n_c(t) \end{bmatrix}. \tag{294}$$

Then,

$$\mathbf{F}(t) = \begin{bmatrix} -k_a & 0 \\ 0 & -k_n \end{bmatrix}, \tag{295}$$

$$\mathbf{G}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{296}$$

$$\mathbf{Q} = \begin{bmatrix} 2k_aP_a & 0 \\ 0 & 2k_nP_n \end{bmatrix}, \tag{297}$$

and

$$\mathbf{C}(t) = [\sqrt{2P}\sin \omega_c t \,\vdots\, 1]. \tag{298}$$

We see that $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{Q}$ are diagonal because of the independence of the message and the colored noise and the fact that each has only one pole. In the general case of independent message and noise we can partition $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{Q}$ and the off-diagonal partitions will be zero.

Modification No. 2: Vector Channels. The next case we need to include to get a general model is one in which we have multiple received waveforms. As we would expect, this extension is straightforward. Assuming m channels, we have a vector observation equation,

$$\mathbf{r}(t) = \mathbf{C}(t)\,\mathbf{x}(t) + \mathbf{w}(t). \tag{299}$$

where $\mathbf{r}(t)$ is m-dimensional. An example illustrates this model.

Example. A simple diversity system is shown in Fig. 6.35. Suppose $a(t)$ is a one-dimensional process. Then $\mathbf{x}(t) = a(t)$. The modulation matrix is $m \times 1$:

$$\mathbf{C}(t) = \begin{bmatrix} c_1(t) \\ c_2(t) \\ \vdots \\ c_m(t) \end{bmatrix}. \tag{300}$$

The channel noises are white with zero-means but may be correlated with one another. This correlation may be time-varying. The resulting covariance matrix is

$$E[\mathbf{w}(t)\,\mathbf{w}^T(u)] \triangleq \mathbf{R}(t)\,\delta(t-u), \tag{301}$$

where $\mathbf{R}(t)$ is positive-definite.

In general, $\mathbf{x}(t)$ is an n-dimensional vector and the channel is m-dimensional so that the modulation matrix is an $m \times n$ matrix. We assume that the channel noise $\mathbf{w}(t)$ and the white process $\mathbf{u}(t)$ which generates the message are uncorrelated.

With these two modifications our model is sufficiently general to include most cases of interest. The next step is to derive a differential equation for the optimum estimate. Before doing that we summarize the important relations.

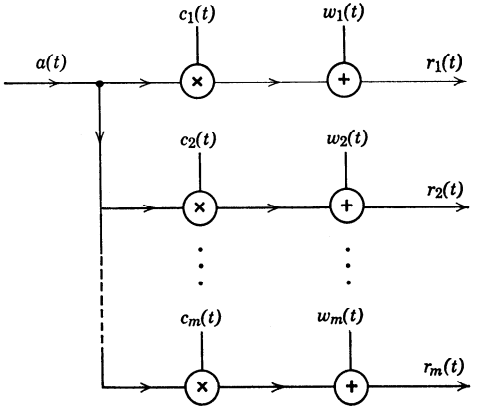


Fig. 6.35 A simple diversity system.

Summary of Model

All processes are assumed to be generated by passing white noise through a linear time-varying system. The processes are described by the vector-differential equation

$$\boxed{\frac{d\mathbf{x}(t)}{dt} = \mathbf{F}(t)\,\mathbf{x}(t) + \mathbf{G}(t)\,\mathbf{u}(t),} \tag{302}$$

where

$$\boxed{E[\mathbf{u}(t)\,\mathbf{u}^T(\tau)] = \mathbf{Q}\,\delta(t-\tau).} \tag{303}$$

and $\mathbf{x}(t_0)$ is specified either as a deterministic vector or as a random vector with known second-moment statistics.

The solution to (302) may be written in terms of a transition matrix:

$$\mathbf{x}(t) = \boldsymbol{\phi}(t, t_0)\,\mathbf{x}(t_0) + \int_{t_0}^{t} \boldsymbol{\phi}(t, \tau)\,\mathbf{G}(\tau)\,\mathbf{u}(\tau)\,d\tau. \tag{304}$$

The output process $\mathbf{y}(t)$ is obtained by a linear transformation of the state vector. It is observed after being corrupted by additive white noise. The received signal $\mathbf{r}(t)$ is described by the equation

$$\boxed{\mathbf{r}(t) = \mathbf{C}(t)\,\mathbf{x}(t) + \mathbf{w}(t).} \tag{305}$$

The measurement noise is white and is described by a covariance matrix:

$$\boxed{E[\mathbf{w}(t)\,\mathbf{w}^T(u)] = \mathbf{R}(t)\,\delta(t-\tau).} \tag{306}$$

Up to this point we have discussed only the second-moment properties of the random processes generated by driving linear dynamic systems with white noise. Clearly, if $\mathbf{u}(t)$ and $\mathbf{w}(t)$ are jointly Gaussian vector processes and if $\mathbf{x}(t_0)$ is a statistically independent Gaussian random vector, then the Gaussian assumption on p. 471 will hold. (The independence of $\mathbf{x}(t_0)$ is only convenient, not necessary.)

The next step is to show how we can modify the optimum linear filtering results we previously obtained to take advantage of this method of representation.

6.3.2 Derivation of Estimator Equations

In this section we want to derive a differential equation whose solution is the minimum mean-square estimate of the message (or messages). We recall that the MMSE estimate of a vector $\mathbf{x}(t)$ is a vector $\hat{\mathbf{x}}(t)$ whose components $\hat{x}_i(t)$ are chosen so that the mean-square error in estimating each

component is minimized. In other words, $E[(\hat{x}_i(t) - x_i(t))^2]$, $i = 1, 2, \ldots, n$ is minimized. This implies that the sum of the mean-square errors, $E\{[\hat{\mathbf{x}}^T(t) - \mathbf{x}^T(t)][\mathbf{x}(t) - \hat{\mathbf{x}}(t)]\}$ is also minimized. The derivation is straightforward but somewhat lengthy. It consists of four parts.

1. Starting with the vector Wiener-Hopf equation (Property 3A–V) for realizable estimation, we derive a differential equation in t, with τ as a parameter, that the optimum filter $\mathbf{h}_o(t, \tau)$ must satisfy. This is (317).
2. Because the optimum estimate $\hat{\mathbf{x}}(t)$ is obtained by passing the received signal into the optimum filter, (317) leads to a differential equation that the optimum estimate must satisfy. This is (320). It turns out that all the coefficients in this equation are known except $\mathbf{h}_o(t, t)$.
3. The next step is to find an expression for $\mathbf{h}_o(t, t)$. Property 4B–V expresses $\mathbf{h}_o(t, t)$ in terms of the error matrix $\xi_P(t)$. Thus we can equally well find an expression for $\xi_P(t)$. To do this we first find a differential equation for the error $\mathbf{x}_\epsilon(t)$. This is (325).
4. Finally, because

$$\xi_P(t) \triangleq E[\mathbf{x}_\epsilon(t)\, \mathbf{x}_\epsilon{}^T(t)], \tag{307}$$

we can use (325) to find a matrix differential equation that $\xi_P(t)$ must satisfy. This is (330). We now carry out these four steps in detail.

Step 1. We start with the integral equation obtained for the optimum finite time point estimator [Property 3A–V, (52)]. We are estimating the entire vector $\mathbf{x}(t)$

$$\mathbf{K}_\mathbf{x}(t, \sigma)\, \mathbf{C}^T(\sigma) = \int_{T_i}^{t} \mathbf{h}_o(t, \tau)\, \mathbf{K}_\mathbf{r}(\tau, \sigma)\, d\tau, \qquad T_i < \sigma < t, \tag{308}$$

where

$$\mathbf{K}_\mathbf{r}(\tau, \sigma) = \mathbf{C}(\tau)\, \mathbf{K}_\mathbf{x}(\tau, \sigma)\mathbf{C}^T(\sigma) + \mathbf{R}(\tau)\, \delta(\tau - \sigma). \tag{309}$$

Differentiating both sides with respect to t, we have

$$\frac{\partial \mathbf{K}_\mathbf{x}(t, \sigma)}{\partial t}\, \mathbf{C}^T(\sigma) = \mathbf{h}_o(t, t)\, \mathbf{K}_\mathbf{r}(t, \sigma) + \int_{T_i}^{t} \frac{\partial \mathbf{h}_o(t, \tau)}{\partial t}\, \mathbf{K}_\mathbf{r}(\tau, \sigma)\, d\tau, \qquad T_i < \sigma < t. \tag{310}$$

First we consider the first term on the right-hand side of (310). For $\sigma < t$ we see from (309) that

$$\mathbf{K}_\mathbf{r}(t, \sigma) = \mathbf{C}(t)[\mathbf{K}_\mathbf{x}(t, \sigma)\mathbf{C}^T(\sigma)], \qquad \sigma < t. \tag{311}$$

The term inside the bracket is just the left-hand side of (308). Therefore,

$$\mathbf{h}_o(t, t)\, \mathbf{K}_\mathbf{r}(t, \sigma) = \int_{T_i}^{t} \mathbf{h}_o(t, t)\, \mathbf{C}(t)\, \mathbf{h}_o(t, \tau)\, \mathbf{K}_\mathbf{r}(\tau, \sigma)\, d\tau, \qquad \sigma < t. \tag{312}$$

Now consider the first term on the left-hand side of (310),

$$\frac{\partial \mathbf{K}_{\mathbf{x}}(t, \sigma)}{\partial t} = E\left[\frac{d\mathbf{x}(t)}{dt}\,\mathbf{x}^T(\sigma)\right]. \tag{313}$$

Using (302), we have

$$\frac{\partial \mathbf{K}_{\mathbf{x}}(t, \sigma)}{\partial t} = \mathbf{F}(t)\,\mathbf{K}_{\mathbf{x}}(t, \sigma) + \mathbf{G}(t)\,\mathbf{K}_{\mathbf{ux}}(t, \sigma), \tag{314}$$

but the second term is zero for $\sigma < t$ [see (266)]. Using (308), we see that

$$\mathbf{F}(t)\,\mathbf{K}_{\mathbf{x}}(t, \sigma)\,\mathbf{C}^T(\sigma) = \int_{T_i}^{t} \mathbf{F}(t)\,\mathbf{h}_o(t, \tau)\,\mathbf{K}_{\mathbf{r}}(\tau, \sigma)\,d\tau. \tag{315}$$

Substituting (315) and (312) into (310), we have

$$\mathbf{0} = \int_{T_i}^{t} \left[-\mathbf{F}(t)\,\mathbf{h}_o(t, \tau) + \mathbf{h}_o(t, t)\,\mathbf{C}(t)\,\mathbf{h}_o(t, \tau) + \frac{\partial \mathbf{h}_o(t, \tau)}{\partial t}\right]\mathbf{K}_{\mathbf{r}}(\tau, \sigma)\,d\tau,$$

$$T_i < \sigma < t. \tag{316}$$

Clearly, if the term in the bracket is zero for all τ, $T_i \leq \tau \leq t$, (316) will be satisfied. Because $\mathbf{R}(t)$ is positive-definite the condition is also necessary; see Problem 6.3.19. Thus the differential equation satisfied by the optimum impulse response is

$$\frac{\partial \mathbf{h}_o(t, \tau)}{\partial t} = \mathbf{F}(t)\,\mathbf{h}_o(t, \tau) - \mathbf{h}_o(t, t)\mathbf{C}(t)\,\mathbf{h}_o(t, \tau). \tag{317}$$

Step 2. The optimum estimate is obtained by passing the input through the optimum filter. Thus

$$\hat{\mathbf{x}}(t) = \int_{T_i}^{t} \mathbf{h}_o(t, \tau)\,\mathbf{r}(\tau)\,d\tau. \tag{318}$$

We assumed in (318) that the MMSE realizable estimate of $\mathbf{x}(T_i) = \mathbf{0}$. Because there is no received data, our estimate at T_i is based on our a priori knowledge. If $\mathbf{x}(T_i)$ is a random variable with a mean-value vector $E[\mathbf{x}(T_i)]$ and a covariance matrix $\mathbf{K}_{\mathbf{x}}(T_i, T_i)$, then the MMSE estimate is

$$\hat{\mathbf{x}}(T_i) = E[\mathbf{x}(T_i)].$$

If $\mathbf{x}(T_i)$ is a deterministic quantity, say $\mathbf{x}_D(T_i)$, then we may treat it as a random variable whose mean equals $\mathbf{x}_D(t)$

$$E[\mathbf{x}(T_i)] \triangleq \mathbf{x}_D(t)$$

and whose covariance matrix $\mathbf{K}_{\mathbf{x}}(T_i, T_i)$ is identically zero.

$$\mathbf{K}_{\mathbf{x}}(T_i, T_i) \triangleq \mathbf{0}.$$