CHAPTER 5

MODERN DESIGN TECHNIQUES

5.1 INTRODUCTION

Modern control theory has made a significant impact on the aircraft industry in recent years. Bryson (e.g., 1985; Ly et al., 1985) pioneered in applying it to aircraft control. Boeing (Gangsaas et al., 1986) has implemented control systems designed using modern techniques, for instance, in the Boeing 767 autopilot. Honeywell has promoted modern robust design (Doyle and Stein, 1981). Linear quadratic methods were used by General Dynamics in the control system of the AFTI/F-16 (*AFWAL-TR-84-308*, 1984).

Therefore, in aircraft control systems design it is essential to have an understanding of modern control theory. Unfortunately, the traditional modern design techniques based on state-variable feedback that are available in current texts are not suitable for aircraft control design. This is due to several things, one of which is their dependence on selecting large numbers of design parameters—namely, the performance index weighting matrices. Any design method for aircraft controls should eliminate the need for this trial-and-error selection. Thus, the techniques by Ly et al. (1985), Gangsaas et al. (1986), Davison and Ferguson (1981), and Moerder and Calise (1985) all rely on modified design techniques that use output feedback or order reduction techniques in conjunction with the minimization of a nonstandard performance index.

In the remainder of the book we focus on modern design techniques that are suitable for use in aircraft control. Included are such approaches as eigenstructure assignment, model-following, dynamic inversion, linear quadratic regulator/loop transfer recovery (LQG/LTR), and linear quadratic (LQ) output feedback design. Each of these techniques has its proponents, and each has its advantages and disadvantages, as we will attempt to demonstrate. We will focus on output feedback

Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems, Third Edition.
Brian L. Stevens, Frank L. Lewis, Eric N. Johnson.

design, with performance criteria that are more general than the usual integral quadratic form. Using this approach it is straightforward to design controllers that have a sensible structure from the point of view of the experience within the aircraft industry without the trial-and-error selection of a large number of design parameters.

Our strategy in the next chapters will be different than in the first part of the book due to the different character of the material to be covered. We will first develop each modern design technique and then present examples showing how it is used in aircraft controls. In several instances we will consider the same examples presented in Chapter 4; this will afford an opportunity to contrast the classical and modern approaches to design. We now discuss some basic philosophy of modern control design.

Limitations of Classical Control

In Chapter 4 we showed how to design aircraft control systems using classical control techniques. The essence of classical design was *successive loop closure* guided by a good deal of intuition and experience that assisted in selecting the control system structure. For instance, we knew it was desirable to provide inner rate feedback loops around a plant to reduce the effect of plant parameter variations. In conjunction with this we used standard compensator structures designed to approximate derivative action to stabilize the system or integral action to eliminate steady-state error.

The one-loop-at-a-time design approach was aided by such tools as root locus, Bode and Nyquist plots, and so on, that enabled us to visualize how the system dynamics were being modified. However, the design procedure became increasingly difficult as more loops were added and did not guarantee success when the dynamics were multivariable, that is, when there were multiple inputs, multiple outputs, or multiple feedback loops.

Philosophy of Modern Control

Two concepts are central to modern control system design. The first is that *the design is based directly on the state-variable model*, which contains more information about the system than the input-output (black box) description. The state-variable model was introduced into system theory, along with matrix algebra, by R. Kalman (1958, 1960). Since we have already seen how to extract state equations from the nonlinear aircraft dynamics and use them for analysis, we are at this point in a good position to use them for control design.

The second central concept is *the expression of performance specifications in terms of a mathematically precise performance criterion* which then yields *matrix equations for the control gains*. These matrix equations are solved using readily available computer software. The classical successive-loop-closure approach means that the control gains are selected individually. In complete contrast, solving matrix equations allows *all the control gains to be computed simultaneously* so that all loops are closed at the same time. This simultaneous design means that we will have greater insight into the design freedom than is possible when the system has more than one input and/or output or multiple control loops. Moreover, using modern control theory we are able to design control systems more quickly and directly than when using classical techniques.

As in classical control, we are able with modern techniques to select the structure of the control system using the intuition developed in the aircraft industry. Thus, it is straightforward to include washout circuits, integral control, and so on. The key to this is the use of *output feedback* design techniques, introduced in Section 5.3 and used throughout the chapter.

The modern control formulation means that the trial and error of one-loop-at-a-time design disappears. Instead, the fundamental engineering decision is *the selection of a suitable performance criterion*. Let us now discuss some important design problems and their associated performance criteria.

Fundamental Design Problems

Pole Placement/Eigenvector Assignment Modern control techniques are available for assigning the poles in multi-input/multi-output (MIMO) systems to desired locations *in one step* by solving equations for the feedback gains. These are called *pole placement techniques*. Once we move away from classical one-loop-at-a-time design and obtain the capability to compute all the feedback gains simultaneously, it will become clear that in the MIMO case *it is possible to do more than simply assign the poles*. In fact, the closed-loop eigenvectors may also be selected within limits.

Desirable pole locations for aircraft design may be found in the military flying qualities specifications (see Section 4.3). However, while discussing flying qualities requirements, we noted that the time response depends not only on the pole locations but also on the zeros of the individual single-input/single-output (SISO) transfer functions, or equivalently on the eigenvectors (see the discussion on system modes in Chapter 3). Thus, the capability of modern control design to select both the closed-loop poles and eigenvectors is relevant in aircraft design.

In this design approach, the performance criterion is to achieve specified pole locations and eigenvectors. We will discuss pole placement/eigenstructure assignment in Section 5.2.

Regulator Problem A fundamental design problem is the *regulator problem*, where it is necessary to regulate the outputs of the system to zero while ensuring that they exhibit desirable time-response characteristics. The regulator problem is important in the design of stability augmentation systems and autopilots.

Stable regulation of systems implies closed-loop stability, but using modern control we may do more than simply ensure stability. To exercise our design freedom, we select as our performance criterion an *integral-squared performance index* (*PI*) similar to those used in classical design (D'Azzo and Houpis, 1988). That is, the squares of the states and inputs are integrated to obtain the PI. The control gains that minimize the PI are found by solving matrix equations using computer programs. Note that if the integral of the squares of the states is made small, then in some sense the states themselves are forced to stay near zero. Selecting different weighting coefficients in the PI for the various state components results in different time-domain behavior in the closed-loop system. Thus, modern control regulator design is fundamentally a *time-domain design technique* useful in shaping the closed-loop response. This is in contrast to classical controls, where most techniques are in the frequency domain.

We will discuss the regulator problem in Section 5.3. A deficiency of the traditional approach to modern regulator design using state feedback and the standard quadratic PI is that, to obtain suitable responses, one must select a large number of design parameters—namely, the PI weighting matrices. To avoid such trial-and-error approaches, we use modified PIs (Section 5.5) that are more suitable for aircraft control system design.

Tracker Problem Another fundamental design problem is the *tracker problem*, where it is desired for an aircraft to follow or track a command signal. The command may be either constant or time varying. This is also referred to as the *servodesign problem*. The tracker problem is important in the design of control augmentation systems, where, for instance, the command signal may be a desired pitch rate or normal acceleration command. The tracker problem also relies on the selection of an integral-squared PI. However, now it is desired to keep the outputs not at zero but near the reference command signals.

The modern control technique we will use for tracker design is not the standard one—it has been modified in several respects to make it more suitable for the purpose of aircraft control design. Specifically, we select a general PI that can easily be modified to attain different performance objectives. The result is a convenient technique for aircraft control design that does not involve the trial-and-error tuning of large numbers of parameters. This design approach is described in Sections 5.4 and 5.5.

Model Following An important approach to control design is *model following*, where it is desired for the aircraft to perform like an ideal model with desirable flying qualities. We have seen in Chapter 4 that one way to specify good flying qualities is to prescribe a low-order model (e.g., with one zero and a complex pole pair) whose response the closed-loop system should match. In this design technique the performance criterion is some measure of the difference between the model and controlled aircraft responses. We cover model-following design in Section 5.6, showing how to design controllers that make the aircraft behave like the model.

As another application, we show that model-following design offers a very straightforward approach to the design of an automatic flare control system. Note that in flare control it is desired for the aircraft to follow an exponential path to a smooth touchdown—here, the model is just the desired trajectory.

Dynamic Inversion The aircraft is fundamentally a nonlinear system. Unfortunately, it is very difficult to design control systems directly for nonlinear systems. Therefore, most aircraft control systems are based on linear systems design at a given operating point. This requires that different controllers be designed for different operating points. Using gain scheduling, the linear controllers are then combined to provide control effectiveness over an operating envelope. One technique that can be applied to nonlinear systems is *dynamic inversion*. This is based on the technique of *feedback linearization*. Dynamic inversion control requires that the controller have a full model of the nonlinearities of the aircraft. Using modern high-power computers, this is possible. In Section 5.8 we cover dynamic inversion, which has gained in popularity in recent years for aircraft control design.

Robust Design It is important to incorporate notions from classical control theory into modern design. Particularly vital is the frequency-domain approach to robustness analysis. However, it is well known that in a multivariable system individual gain and phase margins between different pairs of inputs and outputs mean little from the point of view of overall robustness. Therefore, in Chapter 6 we generalize frequency-domain robustness analysis techniques to MIMO systems using the concept of the *singular value*. There, we also present the LQG/LTR technique, which has recently gained popularity in aircraft controls design.

Observers, Kalman Filter, and Regulators In Chapter 6 we also cover the design of observers and the Kalman filter, which are dynamical systems that estimate the full state from measurements of the outputs. By using feedback of these state estimates in conjunction with the observer dynamics, we are able to design a dynamic linear quadratic regulator, which is just a compensator similar to those obtained using classical techniques. In the modern approach, however, a convenient design method for *multivariable systems* is achieved by solving matrix equations to guarantee specified performance.

Digital Control The control systems of modern aircraft are implemented on digital microprocessors. Examples are the F-15E, F18, and late models of the F-16. The advantages of digital control include the ability to implement complicated multiloop control systems, reprogram the controller gains and structure (e.g., for gain scheduling), obtain redundancy for failure tolerance, and use digital signal processing techniques to filter the control signals.

Digital control design introduces some new problems, such as the need to account for the delays associated with the sampling and hold processes and control computation. Also important is the development of design techniques that overcome the drawbacks associated with z-plane design, where the need arises for extreme accuracy in placing the poles within the unit circle. Implementation problems include accounting for actuator saturation and the effects of computer finite word length, roundoff error, overflow, and so on. We discuss the basics of digital control in Chapter 7.

5.2 ASSIGNMENT OF CLOSED-LOOP DYNAMICS

Classical design techniques such as root locus and Bode analysis are directly applicable only to SISO systems. Using such techniques a single feedback gain may be selected to place the closed-loop poles to guarantee desirable time responses and robustness qualities. In the case of multiple inputs and outputs or multiple control loops, the classical techniques require successive closures of individual loops and involve a significant amount of trial and error.

In the MIMO case it is possible to do more than simply place the poles. This extra freedom is difficult to appreciate from the point of view of classical control theory due to the successive SISO design approach. In this section we want to show how modern control theory can be used in the multivariable case to place the poles as well as to *take advantage of the extra freedom arising from multiple inputs* to assign the closed-loop eigenvectors. This is important since, as we saw in Sections 3.8 and 4.3, the time response of a multivariable system depends not only on the poles but also on the zeros of the individual SISO transfer functions or equivalently on the eigenvectors.

The *eigenstructure assignment* technique discussed in this section offers good insight and is especially useful for the design of *decoupling controllers*, as we will show in an example. As far as obtaining suitable time responses for multivariable systems goes, linear quadratic approaches like those in Sections 5.3 to 5.7 and 6.5 have generally been found more appropriate in the aircraft industry.

We will now discuss some basic feedback concepts from the point of view of modern control theory.

State Feedback and Output Feedback

We have shown that the linearized equations of motion of an aircraft may be written in the state-space form

$$\dot{x} = Ax + Bu \tag{5.2-1}$$

$$y = Cx, \tag{5.2-2}$$

with $x(t) \in \mathbf{R}^n$ the state, $u(t) \in \mathbf{R}^m$ the control input, and $y(t) \in \mathbf{R}^p$ the measured output.

Let us select a feedback control input of the form

$$u = -Kx + v, \tag{5.2-3}$$

where $v(t)$ is an auxiliary input which might be, for instance, the pilot's command and K is an $m \times n$ gain matrix to be determined. This is called a *state-variable feedback* since all of the state components are fed back. The feedback gain K is an $m \times n$ matrix of scalar control gains.

Substituting the control into the state equation yields the closed-loop system

$$\dot{x} = (A - BK)x + Bv \tag{5.2-4}$$

The closed-loop plant matrix is $(A - BK)$, and we would like to select the feedback gain K for good closed-loop performance.

It is a fundamental result of modern control theory that if the system is *controllable*, all of the closed-loop poles may be assigned to desired locations by selection of K. Controllability means that the control input $u(t)$ independently affects all the system modes. It can be tested for by examining the controllability matrix (Kailath, 1980)

$$U = [B \quad AB \quad A^2\dot{B} \quad \cdots \quad A^{n-1}B] \tag{5.2-5}$$

The system is controllable if U has full rank of n, that is, if U has n linearly independent columns. This is equivalent to the nonsingularity of the Gramian UU^T, which is a square $n \times n$ matrix whose determinant can be evaluated.

In the next subsection we will see that if there is more than one control loop, corresponding to more than one control gain, we can not only place the poles but also to a certain extent select the eigenvectors.

Unfortunately, in aircraft control systems it is usually not possible or economically feasible to measure all the states accurately. It is possible to design a dynamic observer or Kalman filter to provide estimates $\hat{x}(t)$ of the states $x(t)$ and then use *feedback of the estimates* by modifying (5.2-3) to read $u = -K\hat{x} + v$. Indeed, we do discuss this approach in Section 6.4, since we need it to cover the LQG/LTR robust design technique in Section 6.5. However, since the aircraft dynamics are nonlinear, all the parameters of any linear observer would need to be gain scheduled. This is inconvenient if the order of the observer is large.

Therefore, to obtain realistic aircraft control schemes, we should feed back not the entire state $x(t)$ but only the measurable outputs $y(t)$. The *output feedback* control law is

$$u = -Ky + v = -KCx + v, \tag{5.2-6}$$

which on substitution into (5.2-1) yields the closed-loop system

$$\dot{x} = (A - BKC)x + Bv \tag{5.2-7}$$

Now the closed-loop plant matrix is $(A - BKC)$. The output feedback matrix K is an $m \times p$ matrix of scalar gains. Thus, since p is generally less than n, there are fewer scalar control gains to select in output feedback design than in state feedback design.

An important advantage of output feedback, as we will see, is that it allows us to *incorporate a compensator of desired form into the feedback system*. In aircraft control, there is a wealth of experience that often dictates the form of the control system. For instance, a washout filter may be required or a PI controller may be needed for zero steady-state error.

Unlike the state feedback case, there is no convenient test to determine for a given system if the closed-loop poles may be independently assigned using output feedback. Pole placement using output feedback is more difficult to accomplish than using state feedback. The basic thrust of this chapter is to investigate the selection of the output feedback gain matrix K to obtain desirable closed-loop characteristics. Note that this will involve more than simply placing the poles, since desirable time responses depend on the poles as well as the zeros of the individual SISO transfer functions. These zeros can also be influenced using feedback if there is more than one input and output.

The gain matrix K is $m \times n$ for state feedback and $m \times p$ for output feedback. In the MIMO case there could be many gain elements, each corresponding to a feedback path. In classical control theory the individual gains must be separately selected using trial-and-error successive loop closure design. By contrast, using modern controls design *all the elements of K are selected simultaneously*. Thus, all the feedback loops in a complicated control system can be closed at the same time with a modern control approach.

We will now discuss the selection of K to yield desired closed-loop poles and eigenvectors—that is, to assign the closed-loop eigenstructure. Both state feedback and output feedback will be considered. First, let us recall the importance of the eigenvectors in the system response.

Modal Decomposition

We have discussed the importance of the system modes in Sections 3.8 and 4.2. Let us now carry that discussion a bit further. Let λ_i be an eigenvalue with right eigenvector v_i and left eigenvector w_i, so that

$$Av_i = \lambda_i v_i, \quad w_i^{\mathrm{T}} A = \lambda_i w_i^{\mathrm{T}} \tag{5.2-8}$$

Since $y = Cx$, we may use the results of Section 3.8 to write the output as

$$y(t) = \sum_{i=1}^{n} \left(w_i^{\mathrm{T}} x_0\right) Cv_i e^{\lambda_i t} + \sum_{i=1}^{n} Cvi \int_0^t e^{\lambda_i (t-\tau)} w_i^{\mathrm{T}} Bu(\tau)\, d\tau \tag{5.2-9}$$

The initial condition is $x(0) = x_0$. This equation is valid when the Jordan form of matrix A is diagonal.

From this equation we may note that Cv_i is a direction in the output space associated with λ_i, while the influence of the control input $u(t)$ on eigenvalue λ_i is determined by $w_i^{\mathrm{T}} B$.

If $Cv_i = 0$, motion in the direction v_i cannot be observed in the output and we say that λ_i is *unobservable*. If $w_i^{\mathrm{T}} B = 0$, the control input $u(t)$ can never contribute to the motion in the direction v_i and we say that λ_i is *uncontrollable*.

Clearly, we may affect the coupling between the inputs, states, and outputs by selecting the vectors v_i and w_i in the closed-loop system; that is, we can influence the *numerators* of the individual SISO transfer functions as well as the poles. To see this clearly, examine the transfer function

$$H(s) = C(sI - A)^{-1} B \tag{5.2-10}$$

Let the Jordan matrix J and matrix of eigenvectors M be

$$J = \mathrm{diag}\,\{\lambda_i\}, \qquad M = [v_1 \quad v_2 \quad \cdots \quad v_n]^{\mathrm{T}}, \tag{5.2-11}$$

so that

$$M^{-1} = [w_1 \quad w_2 \quad \cdots \quad W_n]^{\mathrm{T}} \tag{5.2-12}$$

Now recall that $A = MJM^{-1}$ and use the fact that $(QP)^{-1} = P^{-1}Q^{-1}$ for any two compatible square matrices P and Q to write (5.2-10) as

$$H(s) = CM(sI - J)^{-1}M^{-1}B,$$

or, since $(sI - J)^{-1}$ is diagonal with elements like $1/(s - \lambda_i)$, as

$$H(s) = \sum_{i=1}^{n} \frac{Cv_i w_i^{\mathrm{T}} B}{s - \lambda_i} \tag{5.2-13}$$

This equation gives the partial fraction expansion of $H(s)$ in terms of the eigenstructure of A when A is diagonalizable.

Several things may be said at this point. First, if λ_i is unobservable or uncontrollable, its contribution to the partial fraction expansion of $H(s)$ is zero. In this case we say that the state-space description (5.2-1), (5.2-2) is *not minimal*. Second, the terms Cv_i and $w_i^{\mathrm{T}} B$ determine the residues of the poles and hence the *zeros* of the individual SISO transfer functions in the $p \times m$ matrix $H(s)$. By proposing a technique for selecting the closed-loop eigenvectors by feedback, we are therefore proposing a method of shaping the time response that goes beyond what is possible using only pole placement.

In the next subsections we will show how to design feedback gains that achieve desired closed-loop eigenvectors using both state feedback and output feedback. Meanwhile, in the following example we give some insight on desirable eigenvectors from the point of view of aircraft behavior, recalling some results from Sections 3.8 and 4.2. We also show that the eigenvectors may be selected to obtain *decoupling between the system modes*.

Example 5.2-1: Selecting Eigenvectors for Decoupling In Sections 3.8 and 4.2 we studied the aircraft longitudinal and lateral modes, showing which states are involved in each one. To ensure that the controlled aircraft exhibits suitable flying qualities, we should take care to design the control system so that this basic modal structure is preserved (Sobel and Shapiro, 1985; Andry et al., 1983).

In this example we idealize the findings of Sections 3.8 and 4.2 a bit. That is, we make more categorical statements about the mode couplings in order to obtain concrete design objectives.

(a) ***Longitudinal Axis.*** Assuming that the state equations are augmented by a simple-lag elevator actuator model, in the linearized perturbed longitudinal equations of an aircraft the state can be taken as

$$x = \begin{bmatrix} \alpha \\ q \\ \theta \\ v_T \\ \delta_e \end{bmatrix} \tag{1}$$

The states are angle of attack, α, pitch rate, q, pitch angle, θ, total velocity, v_T, and elevator actuator state, δ_e, the elevator deflection. We have ordered the states this way to make the upcoming discussion clearer [see (2)].

The short-period mode is due primarily to a coupling of energy between α and q. The phugoid mode is due primarily to a coupling between θ and v_T. It is desirable for the forward velocity to be unaffected by short-period oscillations, while the phugoid oscillations should have no influence on angle of attack.

To achieve this behavior, we could select the closed-loop eigenvectors as

$$v_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ x \end{bmatrix} + j \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \\ x \end{bmatrix}, \quad v_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ x \end{bmatrix} + j \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \\ x \end{bmatrix}, \tag{2}$$

with v_2 and v_4 the complex conjugates of v_1 and v_3, respectively. Components whose values we do not care about are denoted by x. Then one oscillatory mode, the one with directions specified by v_1 and v_2, will involve the first two components of the state vector but will not inject energy into components 3 and 4. This will be a "good" short-period mode. Similarly, the phugoid mode, described by the eigenvectors v_3 and v_4, will involve components 3 and 4 but will not affect components 1 and 2.

A good design will have these closed-loop eigenvectors. The closed-loop poles should also be specified to attain the desired frequency and damping of the short-period and phugoid modes. The former will be defined by λ_1 and λ_2, while the latter will be determined by λ_3 and λ_4.

It may be necessary to modify the A-matrix to a form that involves the nondimensional time to make the modal coupling of the eigenvectors more apparent (McRuer et al., 1973). That is, the modified A-matrix should have the eigenvectors in (2).

(b) *Lateral Axis.* In the linearized perturbed lateral equations of an aircraft the state can be taken as

$$x = \begin{bmatrix} r \\ \beta \\ p \\ \phi \\ \delta_r \\ \delta_a \end{bmatrix}, \tag{3}$$

where we have assumed first-order lags for the rudder and aileron actuators. The states are yaw rate, r, sideslip angle, β, roll rate, p, bank angle, ϕ, and the actuator states, δ_r, the rudder deflection, and δ_a, the aileron deflection.

Roll commands should not excite the dutch roll mode. Thus let us associate r and β in the closed-loop system with the dutch roll mode (see Section 4.2). Then the roll subsidence mode, which involves p, should not influence r and β. Similarly, the dutch roll oscillation should have no effect on roll rate or bank angle. Desirable eigenvectors to achieve this decoupling between modes are given by

$$\boldsymbol{v}_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ x \\ x \end{bmatrix} + j \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \\ x \\ x \end{bmatrix}, \quad \boldsymbol{v}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ x \\ x \\ x \end{bmatrix}, \tag{4}$$

with $\boldsymbol{v}_2$ the complex conjugate of $\boldsymbol{v}_1$ and x denoting entries whose values we are not concerned about. The closed-loop poles should also be selected for desirable time response: λ_1 and λ_2 for the dutch roll mode and λ_3 for the roll subsidence mode. The desired closed-loop spiral mode may be selected as λ_4. ■

Eigenstructure Assignment by Full State Feedback

Now that we have seen what the eigenvectors mean from the point of view of the aircraft behavior, we will discuss the assignment of both the closed-loop poles and eigenvectors, first by full state feedback and then in the next subsection by output feedback. This represents an extension of classical control theory in several ways. First, we are able to deal in a natural fashion with MIMO systems, *selecting all the control gains simultaneously for suitable performance*. Second, we will be able to use the extra freedom in systems with more than one input and output to assign the eigenvectors as well as the poles, thus directly influencing the *zeros* of the individual SISO transfer functions. Third, we will be able to address the problem of *decoupling of the modes* through considerations like those in Example 5.2-1.

Matrix Equation for Eigenstructure Assignment For ease of presentation we will assume that B and C have full rank m and p, respectively. Our discussion will be based on the polynomial matrix

$$C(s) = [sI - A \quad B], \tag{5.2-14}$$

with s a complex variable and A and B the system plant and input matrices.

In this subsection we follow Moore (1975) and consider full state feedback of the form

$$u = -Kx \tag{5.2-15}$$

Under the influence of this control input the closed-loop system becomes

$$\dot{x} = (A - BK)x \tag{5.2-16}$$

To select K so that a desired eigenvalue λ_i and associated eigenvector v_i are assigned to the closed-loop system, suppose that we can find a vector $u_i \in \mathbf{R}^m$ to satisfy the equation

$$[\lambda_i I - A \quad B] \begin{bmatrix} v_i \\ u_i \end{bmatrix} = 0 \tag{5.2-17}$$

Now, choose the feedback gain K to satisfy

$$K v_i = u_i \tag{5.2-18}$$

Using the last two equations, we may write

$$0 = (\lambda_i I - A) v_i + B u_i \tag{5.2-19}$$

$$0 = [\lambda_i I - (A - BK)] v_i, \tag{5.2-20}$$

so that according to (5.2-8), v_i is assigned as a closed-loop eigenvector for eigenvalue λ_i.

As Cv_i was shown in the preceding subsection to be a direction in the output space $\mathbf{R}^p$ associated with v_i, so u_i is the associated direction in the input space $\mathbf{R}^m$. That is, motions of $u(t)$ in the direction of u_i will cause motions of $x(t)$ in the direction of v_i, resulting in motions of $y(t)$ in the direction of Cv_i.

To complete the picture, suppose that n eigenvalues λ_i and associated eigenvectors v_i are chosen and in each case we have found a vector u_i that satisfies (5.2-17). Then we may define K by

$$K[v_1 \quad v_2 \quad \cdots \quad v_n] = [u_1 \quad u_2 \quad \cdots \quad v_n] \tag{5.2-21}$$

or by appropriate definition of the matrices V and U,

$$KV = U \tag{5.2-22}$$

Then, for each value of $i = 1, \ldots, n$, (5.2-20) will hold, so that each λ_i will be assigned as a closed-loop pole with associated eigenvector v_i. This is the design technique for eigenstructure assignment using full state feedback. It remains only to discuss a few points.

Design Considerations Since, by definition, the closed-loop eigenvectors must be linearly independent, it is necessary to select v_i as linearly independent vectors. Then (5.2-22) may be solved for K to give

$$K = UV^{-1} \tag{5.2-23}$$

Another issue is that the closed-loop system and feedback gain must be real and not complex. Thus, if a complex closed-loop pole λ_i is selected, it is also necessary to select as a closed-loop pole its complex conjugate λ_i^*. Moreover, if v_i is to be the closed-loop eigenvector associated with a complex pole λ_i, then in order for (5.2-21) to have a real solution K, it is necessary to select v_i^* (i.e., the complex conjugate of v_i) as the eigenvector for λ_i^*.

To see that under these circumstances (5.2-22) indeed has a real solution K, note first that if u_i solves (5.2-19) for a given λ_i and v_i, then u_i^* solves the equation for their complex conjugates. Therefore, if $u_i = u_R + ju_I$ and $v_i = v_R + jv_I$, then to assign the desired eigenstructure, K must satisfy

$$K\begin{bmatrix} v_R + jv_I & v_R - jv_I \end{bmatrix} = \begin{bmatrix} u_R + ju_I & u_R - ju_I \end{bmatrix} \tag{5.2-24}$$

Postmultiplying both sides of this equation by

$$M = \frac{1}{2}\begin{bmatrix} 1 & -j \\ 1 & j \end{bmatrix},$$

it is seen that this equation is equivalent to the real equation

$$K[v_R \quad v_I] = [u_R \quad u_I], \tag{5.2-25}$$

which clearly has as a solution a real gain matrix K. Thus, if v_i is complex, then to obtain a real value for K it is only necessary to use not v_i and v_i^* (respectively u_i and u_i^*) in (5.2-21) but the real and imaginary parts of v_i (respectively u_i).

Finally, we must investigate the conditions for existence of a solution to (5.2-17). It is unfortunately not usually possible to specify independently an arbitrary λ_i and v_i and obtain a solution u_i to this equation. Indeed, assuming that λ_i is not an open-loop pole, we have

$$v_i = -(\lambda_i I - A)^{-1} B u_i \tag{5.2-26}$$

Thus, for the existence of a solution u_i, the desired v_i must be a linear combination of the m columns of the linear operator

$$L_i = (\lambda_i I - A)^{-1} B \tag{5.2-27}$$

Since B has full rank m by assumption, the matrix L_i also has rank m. Thus, v_i must lie in an m-dimensional subspace of $\mathbf{R}^n$ that depends on the choice of λ_i. This means that we have *m degrees of freedom in selecting the closed-loop eigenvector* v_i once λ_i has been selected.

This last point is the crucial difference between classical SISO design and multivariable eigenstructure assignment. If $m = 1$, which corresponds to the single-input case, then eigenvector v_i has only one degree of freedom once the desired eigenvalue λ_i has been selected; that is, there is no additional freedom to choose the eigenvector. However, in the multi-input case where $m > 1$, we can have additional freedom to specify the internal structure of the closed-loop system by selecting m degrees of freedom of v_i arbitrarily. In the preceding subsection we have seen the importance of this in terms of design performance.

The successive-loop-closure approach of classical control, where only one feedback gain is selected at a time, obscures the extra design freedom arising from multiple inputs. In modern control, where all gains are selected simultaneously, this freedom is clearly revealed.

Design Procedure The following design procedure for eigenstructure assignment is suggested. For a desired closed-loop pole/vector pair of λ_i and v_i^d, solve the equation

$$\begin{bmatrix} \lambda_i I - A & B \\ d & 0 \end{bmatrix} \begin{bmatrix} v_i \\ u_i \end{bmatrix} = \begin{bmatrix} 0 \\ v_i^d \end{bmatrix} \tag{5.2-28}$$

for u_i and the *achievable eigenvector* v_i. Repeat for $i = 1, \ldots, n$ to select n closed-loop poles. If the v_is are not linearly independent, modify the choices for λ_i and/or v_i^d and repeat. Finally, determine the required state-variable feedback gain K using (5.2-23).

The design matrix D may be chosen for several different design objectives:

1. $D = I$. This is the case where the desired vectors v_i^d are eigenvectors (as in Example 5.2-1).
2. $D = C$. This is the case where the desired vectors are directions in the output space $\mathbf{R}^p$, so that we desire $Cv_i = v_i^d$.
3. If certain components of v_i are of no concern (see Example 5.2-1), the corresponding columns of D should be selected as zero. The remaining columns should be selected as columns from the $d \times d$ identify matrix, with d the number of rows of D. The elements of v_i that they multiply should be as specified by the elements of v_i^d. We illustrate further in Example 5.2-3.

We have seen that (5.2-28) may not have an exact solution v_i, u_i. It is necessary to find a solution so that (1) the first n equations hold exactly [i.e., (5.2-19) must hold exactly] and (2) the second block equation $Dv_i = v_i^d$ holds as closely as possible (then our design objectives are most closely matched). Subroutine LLBQF in the IMSL library (IMSL) allows us to do this. It gives a least-squares solution to the second equation in the sense that $\|Dv_i = v_i^d\|^2$ is minimized over all possible v_i for which there exists a u_i that satisfies (5.2-19) (where $\|w\|$ is the Euclidean norm of vector w).

An interactive design technique is suggested wherein:

1. Given the desired λ_i and v_i^d, (5.2-28) is solved for the pair v_i, u_i meeting the requirements above.
2. The achievable eigenvector v_i is compared with the desired eigenvector and if it is unsatisfactory either v_i^d or λ_i may be modified and step 1 repeated.

Eigenstructure Assignment by Output Feedback

In an aircraft control system, all of the states are not generally available for measurement. Instead, only selected outputs are available for control purposes. It is not difficult to modify the eigenstructure assignment technique so that the admissible controls are of the form

$$u = -Ky, \tag{5.2-29}$$

with output $y(t) \in \mathbf{R}^p$ given by (5.2-2) (Srinathkumar, 1978). In this case, we will show that p eigenvalues may easily be assigned, with m degrees of freedom in the choice of the associated eigenvectors.

Matrix Equation for Eigenstructure Assignment In the case of output feedback the closed-loop system is

$$\dot{x} = (A - BKC)x \tag{5.2-30}$$

and it is only necessary to replace (5.2-18) by

$$KCv_i = u_i \tag{5.2-31}$$

Then, according to (5.2-17),

$$0 = (\lambda_i I - (A - BKC))v_i, \tag{5.2-32}$$

so that v_i is assigned as a closed-loop eigenvector for eigenvalue λ_i. In this case, (5.2-21) for K is replaced by

$$KC[v_1 \quad v_2 \quad \cdots \quad v_r] = [u_1 \quad u_2 \quad \cdots \quad u_r], \tag{5.2-33}$$

where r is the number of closed-loop eigenvalues selected.

If $r = p$ and the vectors Cv_i are linearly independent, we may define

$$V = [v_1 \quad v_2 \quad \cdots \quad v_r], \quad U = [u_1 \quad u_2 \quad \cdots \quad u_r] \tag{5.2-34}$$

and solve for K using

$$K = U(CV)^{-1} \tag{5.2-35}$$

Thus, it is clear what is lost by using incomplete state information for feedback purposes, for we can in general no longer assign n poles arbitrarily.

Extensions What we have demonstrated is a technique for assigning by output feedback p closed-loop poles, with m degrees of freedom in specifying the components of each associated closed-loop eigenvector (where m is the number of inputs).

If $m \leq p$, so that the number of inputs is greater than or equal to the number of outputs, the technique just presented is suitable. However, if $m > p$, we must use the technique on the "dual" system. That is, (A, B, C) is replaced by $(A^{\mathrm{T}}, C^{\mathrm{T}}, B^{\mathrm{T}})$ and the design is performed to find K^{T}. In this case we may assign m closed-loop poles, with p degrees of freedom in assigning the associated eigenvectors.

A problem with eigenstructure assignment using output feedback is that it is not possible to tell what happens to the $n - p$ poles that are not assigned. Indeed, some of them may become unstable, even though the original plant was stable. If this occurs, or if some closed-loop poles are too lightly damped, the design should be repeated using different values for λ_i or v_i^d. Generally, it is found that if one does not ask for too much in terms of modifying the original plant behavior, that is, if most of the desired closed-loop poles are not too different from the open-loop poles, instability of the unassigned poles is not a problem (as long as they are open-loop stable).

Srinathkumar shows that it is possible to assign an almost arbitrary set of min $(n, m + p - 1)$ eigenvalues, but we will not go into details here. Kwon and Youn (1987) show that it may be possible to assign $m + p$ poles in some examples.

The next concept is quite important, so we will illustrate it by an example.

Example 5.2-2: Eigenstructure Assignment Using Dynamic Regulator We have shown how to select constant-feedback gains to assign the closed-loop eigenstructure. However, it is possible to obtain a desired modal structure by using a dynamic compensator. This example shows how to design a dynamic compensator for eigenstructure assignment.

Consider the plant

$$\dot{x} = Ax + Bu \tag{1}$$

$$y = Cx \tag{2}$$

with the regulator dynamics defined by

$$\dot{z} = Fz + Gy \tag{3}$$

$$u = Hz + Jy \tag{4}$$

This corresponds to the situation in Figure 5.2-1. Matrices F, G, H, and J are to be selected to yield a desired closed-loop eigenstructure.

Show that by defining the augmented plant, input, and output matrices

$$\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} B & 0 \\ 0 & I \end{bmatrix}, \quad \begin{bmatrix} C & 0 \\ 0 & I \end{bmatrix} \tag{5}$$

and the gain matrix

$$K = \begin{bmatrix} -J & -H \\ -G & -F \end{bmatrix} \tag{6}$$

the problem of determining F, G, H, and J to yield desired closed-loop poles λ_i and eigenvectors v_i^d may be solved by using the techniques of this section to determine K.

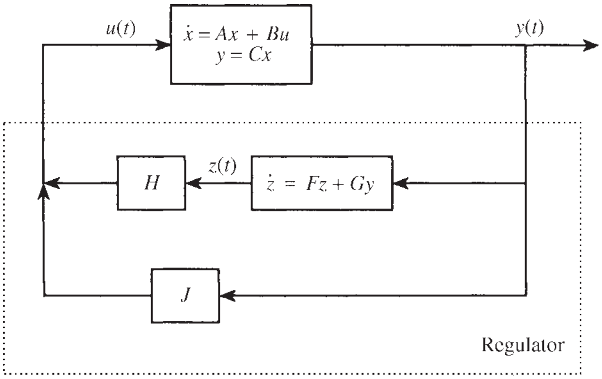


Figure 5.2-1 Plant with regulator.

Note: A problem with this approach is that the regulator matrix F cannot be guaranteed stable. An alternative approach to regulator/observer design is given by Andry et al. (1984). ■

Example 5.2-3: Eigenstructure Design of Longitudinal Pitch Pointing Control. This example is taken from the work of Sobel and Shapiro (1985). A linearized model of the short-period dynamics of an advanced (CCV-type) fighter aircraft is given. These dynamics are augmented by elevator and flaperon actuator dynamics given by the simplified model $20/(s+20)$ so that the state vector is

$$x = \begin{bmatrix} \alpha \\ q \\ \gamma \\ \delta_e \\ \delta_f \end{bmatrix}, \tag{1}$$

where the state components are, respectively, angle of attack, pitch rate, flight-path angle, elevator deflection, and flaperon deflection. The control inputs are elevator and flaperon commands so that

$$u = \begin{bmatrix} \delta_{e_c} \\ \delta_{f_c} \end{bmatrix} \tag{2}$$

The plant and control matrices are

$$A = \begin{bmatrix} -1.341 & 0.9933 & 0 & -0.1689 & -0.2518 \\ 43.223 & -0.8693 & 0 & -17.251 & -1.5766 \\ 1.341 & 0.0067 & 0 & 0.1689 & 0.2518 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix} \tag{3}$$

and the open-loop eigenvalues are

$$\begin{aligned} &\left.\begin{aligned}\lambda_1 &= 5.452\\ \lambda_2 &= -7.662\end{aligned}\right\} && \text{unstable short-period mode}\\ &\lambda_3 = 0.0 && \text{pitch-attitude mode}\\ &\lambda_4 = -20 && \text{elevator actuator mode}\\ &\lambda_5 = -20 && \text{flaperon actuator mode}\end{aligned} \tag{4}$$

The measured output available for control purposes is

$$y = \begin{bmatrix} q \\ n_{zp} \\ \gamma \\ \delta_e \\ \delta_f \end{bmatrix}, \tag{5}$$

where n_{zp} is normal acceleration at the pilot's station. The altitude rate $\dot{h}$ is obtained from the air data computer and the flight-path angle is then computed using

$$\gamma = \frac{\dot{h}}{TAS} \tag{6}$$

with T as the true airspeed. The control surface deflections are measured using linear variable differential transformers (LVDTs). The relation between $y(t)$ and $x(t)$ is given by

$$y = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 47.76 & -0.268 & 0 & -4.56 & 4.45 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x \equiv Cx \tag{7}$$

Since there are five outputs and two control inputs, we may place all the closed-loop poles as well as assign the eigenvectors within two-dimensional subspaces. This roughly corresponds to selecting two components of each eigenvector arbitrarily.

The desired closed-loop short-period poles are chosen to meet military specifications for category A, level-1 flight (*Mil. Spec. 1797*, 1987) (see Section 4.3). Thus, the desired short-period damping ratio and frequency are 0.8 and 7 rad/s, respectively.

For stability, we specify that the desired closed-loop pitch-attitude mode should decay exponentially with a time constant of 1, so that the pole should be at $s = -1$. The actuator poles should be near -20; however, selecting repeated poles can yield problems with the design algorithm. The desired eigenvalues are thus selected as

$$\left.\begin{aligned}\lambda_1 &= -5.6 + j4.2\\ \lambda_2 &= -5.6 - j4.2\end{aligned}\right\}\ \text{short-period mode}$$

$$\begin{aligned}\lambda_3 &= -1.0 &&\text{pitch-attitude mode}\\ \lambda_4 &= -19.0 &&\text{elevator actuator mode}\\ \lambda_5 &= -19.5 &&\text{flaperon actuator mode}\end{aligned}\tag{8}$$

In pitch pointing, the control objective is to allow pitch-attitude control while maintaining constant flight-path angle. To achieve this we select the desired closed-loop eigenvectors to decouple pitch-rate and flight-path angle. Thus, an attitude command should be prevented from causing a significant flight-path change. The desired closed-loop eigenvectors are shown in Table 5.2-1, where x denotes elements of no concern to us. Recall that α and q are associated with the short-period mode.

We now discuss the design procedure and the selection of the D-matrix in the design equation (5.2-28). We must determine the vectors v_i and u_i for use in (5.2-33) to solve for the feedback gain matrix K. To accomplish this, first consider the desired structure of the short-period mode. According to Table 5.2-1, the required short-period eigenvectors have two "don't care" entries. Define v_1^d in terms of the required eigenvector as

$$v_1^d = \begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\end{bmatrix}\left[\begin{bmatrix}1\\-1\\0\\x\\x\end{bmatrix}+j\begin{bmatrix}-1\\1\\0\\x\\x\end{bmatrix}\right]=\begin{bmatrix}1\\-1\\0\end{bmatrix}+j\begin{bmatrix}-1\\1\\0\end{bmatrix}\tag{9}$$

to be the desired vector associated with $\lambda_1 = -5.6 + j4.2$ and select D as the 3×6 coefficient matrix in (9). Then (5.2-28) may be solved for v_1 and u_1. Then the vectors

TABLE 5.2-1 Desired and Achievable Eigenvectors

Desired Eigenvectors							*Achievable Eigenvectors*					
1		−1	x	x	x	α	−0.93		1	−1	−0.051	0.01
−1		1	0	x	x	q	1		−9.5	0	1.07	0.06
0	$+j$	0	1	x	x	γ	0	$+j$	0	1	−0.006	−0.014
x		x	x	1	x	δ_e	−5.13		0.129	−2.8	1	0
x		x	x	x	1	δ_f	8.36		−5.16	3.23	0	1
α/q Short period			γ	δ_e	δ_f		α/q Short period			γ	δ_e	δ_f

associated with $\lambda_2 = \lambda_1^*$ are $v_2 = v_1^*$, $u_2 = u_1^*$. The achievable eigenvectors v_1 and v_2 associated with the short-period mode are shown in Table 5.2-1.

To determine whether the results to this point are satisfactory, the achievable eigenvectors v_1 and v_2 are compared with the desired eigenvectors. They are satisfactory since there is no coupling to state component 3. Note that although we attempted to select three components of the eigenvectors knowing that there are only two degrees of freedom in this selection, we have nevertheless been fortunate in attaining our design objectives. Had we not been so lucky, it would have been necessary to try different desired eigenvectors or else slightly different values for the closed-loop poles.

Moving on to the desired structure of λ_3, examine Table 5.2-1 to define

$$v_3^d = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ 0 \\ 1 \\ x \\ x \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{10}$$

to be the desired vector associated with $\lambda_3 = -1.0$ and select D as the 2 × 6 coefficient matrix in (10). Then (5.2-28) may be solved for v_3 and u_3. The result is the achievable eigenvector v_3 shown in Table 5.2-1; again, it is suitable.

To design for the desired structure of λ_4, examine Table 5.2-1 to define

$$v_4^d = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ x \\ x \\ 1 \\ x \end{bmatrix} = 1 \tag{11}$$

to be the desired vector associated with $\lambda_4 = -19.0$ and select D as the 1 × 6 coefficient matrix in (11). Then (5.2-28) may be solved for v_4 and u_4. The results are in the table. Similar procedures apply for λ_5.

Now that all the requisite vectors v_i and u_i, $i = 1, 2, 3, 4, 5$, have been computed, they are used, along with the C-matrix from (7), to solve for the feedback gain using (5.2-33). The result is

$$K = \begin{bmatrix} -0.931 & -0.149 & -3.25 & -0.153 & 0.747 \\ 0.954 & 0.210 & 6.10 & 0.537 & -1.04 \end{bmatrix} \tag{12}$$

To check the design, a computer simulation was performed. The closed-loop system was excited with an initial condition of 0.2 rad in angle of attack. Note from Figure 5.2-2 that this excited the short-period mode but had negligible effect on the flight-path angle.

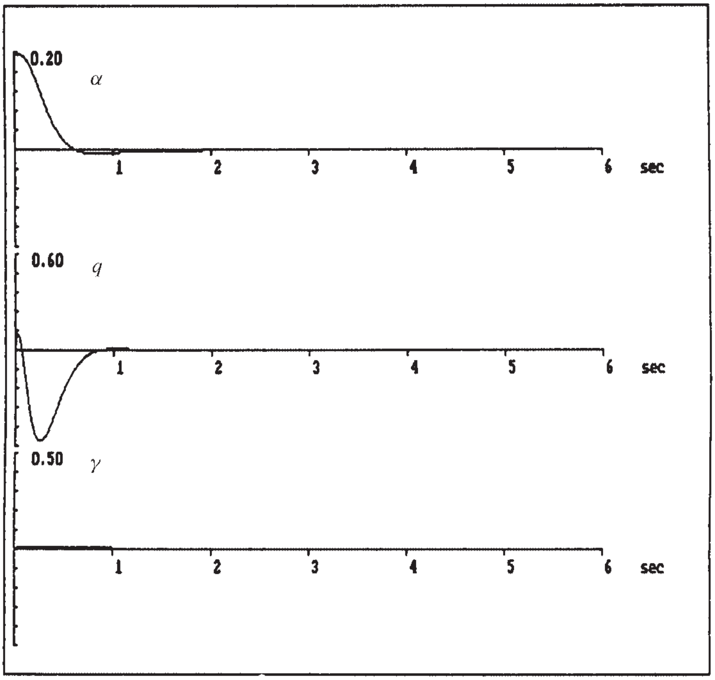


Figure 5.2-2 Closed-loop response to angle-of-attack initial condition. ■

5.3 LINEAR QUADRATIC REGULATOR WITH OUTPUT FEEDBACK

Our objective in this section is to show how to use modern techniques to design stability augmentation systems (SASs) and autopilots. This is accomplished by regulating certain states of the aircraft to zero while obtaining desirable closed-loop response characteristics. It involves the problem of stabilizing the aircraft by placing the closed-loop poles at desirable locations.

Using classical control theory, we were forced to take a one-loop-at-a-time approach to designing multivariable SASs and autopilots. In this section we will select a performance criterion that reflects our concern with closed-loop stability and good time responses and then derive matrix equations that may be solved for *all the control gains simultaneously*. These matrix equations are solved using digital computer programs (see Appendix B). This approach thus closes all the loops simultaneously and results in a simplified design strategy for MIMO systems or SISO systems with multiple feedback loops.

Once the performance criterion has been selected, the control gains are explicitly computed by marix design equations, and closed-loop stability will generally be guaranteed. This means that *the engineering judgment in modern control enters into the selection of the performance criterion*. Different criteria will result in different closed-loop time responses and robustness properties.

We assume the plant is given by the linear time-invariant state-variable model

$$\dot{x} = Ax + Bu \tag{5.3-1}$$

$$y = Cx, \tag{5.3-2}$$

with $x(t) \in \mathbf{R}^n$ the state, $u(t) \in \mathbf{R}^m$ the control input, and $y(t) \in \mathbf{R}^p$ the measured output. The controls will be output feedbacks of the form

$$u = -Ky, \tag{5.3-3}$$

where K is an $m \times p$ matrix of constant-feedback coefficients to be determined by the design procedure. Since the regulator problem only involves stabilizing the aircraft and inducing good closed-loop time responses, $u(t)$ will be taken as a pure feedback with no auxiliary input (see Section 5.2).

As we will see in Section 5.4, output feedback will allow us to design aircraft controllers of any desired structure. This is one reason for preferring it over full state feedback.

In the regulator problem, we are interested in obtaining good time responses as well as in the stability of the closed-loop system. Therefore, we will select a performance criterion *in the time domain*. Let us now present this criterion.

Quadratic Performance Index

The objective of state regulation for the aircraft is to drive any initial condition error to zero, thus guaranteeing stability. This may be achieved by selecting the control input $u(t)$ to minimize a quadratic *cost* or *performance index* (*PI*) of the type

$$J = \tfrac{1}{2}\int_0^\infty (x^{\mathrm{T}}Qx + u^{\mathrm{T}}Ru)\,dt, \tag{5.3-4}$$

where Q and R are symmetric positive-semidefinite *weighting matrices*. Positive semidefiniteness of a square matrix M (denoted $M \geq 0$) is equivalent to all its eigenvalues being nonnegative and also to the requirement that the quadratic form $x^{\mathrm{T}}Mx$ be nonnegative for all vectors x. Therefore, the definiteness assumptions on Q and R guarantee that J is nonnegative and lead to a sensible minimization problem. This quadratic PI is a vector version of an integral-squared PI of the sort used in classical control (D'Azzo and Houpis, 1988).

To understand the motivation for the choice of (5.3-4), consider the following. If the square root $\sqrt{M}$ of a positive-semidefinite matrix M is defined by

$$M = \sqrt{M}^{T}\sqrt{M}, \tag{5.3-5}$$

we may write (5.3-4) as

$$J = \tfrac{1}{2}\int_0^\infty \left(\left\|\sqrt{Q}x\right\|^2 + \left\|\sqrt{R}u\right\|^2\right)dt, \tag{5.3-6}$$

with $\|w\|$ the Euclidean norm of a vector w (i.e., $\|w\|^2 = w^{\mathrm{T}}w$). If we are able to select the control input $u(t)$ so that J takes on a minimum finite value, certainly the integrand must become zero for large time. This means that both the linear combination $\sqrt{Q}x(t)$ of the states and the linear combination $\sqrt{R}u(t)$ of the controls must go to zero. In different designs we may select Q and R for different performance requirements, corresponding to specified functions of the state and input. In particular, if Q and R are both chosen nonsingular, the entire state vector $x(t)$ and all the controls $u(t)$ will go to zero with time if J has a finite value.

Since a bounded value for J will guarantee that $\sqrt{Q}x(t)$ and $\sqrt{R}u(t)$ go to zero with time, this formulation for the PI is appropriate for the regulator problem as any initial condition errors will be driven to zero.

If the state vector $x(t)$ consists of capacitor voltages $v(t)$ and inductor currents $i(t)$, then $\|x\|^2$ will contain terms like $v^2(t)$ and $i^2(t)$. Similarly, if velocity $s(t)$ is a state component, $\|x\|^2$ will contain terms like $s^2(t)$. Therefore, the minimization of the PI (5.3.4) is a generalized *minimum-energy* problem. We are concerned with minimizing the energy in the states without using too much control energy.

The relative magnitudes of Q and R may be selected to trade off requirements on the smallness of the state against requirements on the smallness of the input. For instance, a larger control-weighting matrix R will make it necessary for $u(t)$ to be smaller to ensure that $\sqrt{R}u(t)$ is near zero. We say that a larger R *penalizes* the controls more, so that they will be smaller in norm relative to the state vector. On the other hand, to make $x(t)$ go to zero more quickly with time, we may select a larger Q.

As a final remark on the PI, we will see that the positions of the closed-loop poles depend on the choices for the weighting matrices Q and R. That is, Q and R may be chosen to yield good time responses in the closed-loop system.

Let us now derive matrix design equations that may be used to solve for the control gain K that minimizes the PI. The result will be the design equations in Table 5.3-1. Software to solve these equations for K is described in Appendix B.

Solution of the LQR Problem

The LQR problem with output feedback is the following. Given the linear system (5.3-1), (5.3-2), find the feedback coefficient matrix K in the control input (5.3-3) that minimizes the value of the quadratic PI (5.3-4). In contrast with most of the classical control techniques given in earlier chapters, this is a *time-domain* design technique.

By substituting the control (5.3-3) into (5.3-1) the closed-loop system equations are found to be

$$\dot{x} = (A - BKC)x \equiv A_c x \tag{5.3-7}$$

The PI may be expressed in terms of K as

$$J = \tfrac{1}{2}\int_0^\infty x^{\mathrm{T}}(Q + C^{\mathrm{T}}K^{T}RKC)x\,dt \tag{5.3-8}$$

The design problem is now to select the gain K so that J is minimized subject to the dynamical constraint (5.3-7).

This *dynamical* optimization problem may be converted into an equivalent *static* one that is easier to solve as follows. Suppose that we can find a constant, symmetric, positive-semidefinite matrix P so that

$$\frac{d}{dt}(x^{\mathrm{T}}Px) = -x^{\mathrm{T}}(Q + C^{\mathrm{T}}K^{\mathrm{T}}RKC)x \tag{5.3-9}$$

Then J may be written as

$$J = \tfrac{1}{2}x^{\mathrm{T}}(0)Px(0) - \tfrac{1}{2}\lim_{1\to\infty} x^{\mathrm{T}}(t)Px(t) \tag{5.3-10}$$

Assuming that the closed-loop system is asymptotically stable so that $x(t)$ vanishes with time, this becomes

$$J = \tfrac{1}{2}x^{\mathrm{T}}(0)Px(0) \tag{5.3-11}$$

If P satisfies (5.3-9), we may use (5.3-7) to see that

$$\begin{aligned} -x^{\mathrm{T}}(Q + C^{\mathrm{T}}K^{\mathrm{T}}RKC)x &= \frac{d}{dt}(x^{\mathrm{T}}Px) = \dot{x}^{\mathrm{T}}Px + x^{\mathrm{T}}P\dot{x} \\ &= x^{\mathrm{T}}(A_c^TP + PA_c)x \end{aligned} \tag{5.3-12}$$

Since this must hold for all initial conditions and hence for all state trajectories $x(t)$, we may write

$$g = A_c^{\mathrm{T}}P + PA_c + C^{\mathrm{T}}K^{\mathrm{T}}RKC + Q = 0 \tag{5.3-13}$$

If K and Q are given and P is to be solved for, this is called a *Lyapunov equation*. (A Lyapunov equation is a symmetric linear matrix equation. Note that the equation does not change if its transpose is taken.)

In summary, for any fixed feedback matrix K, if there exists a constant, symmetric, positive-semidefinite matrix P that satisfies (5.3-13) and if the closed-loop system is stable, the cost J is given in terms of P by (5.3-11). This is an important result in that the $n \times n$ auxiliary matrix P is independent of the state. Given a feedback matrix K, P may be computed from the Lyapunov equation (5.3-13). Then only the initial condition $x(0)$ is required to compute the closed-loop cost under the influence of the feedback control $u = -Ky$ *before we actually apply it.*

It is now necessary to use this result to compute the gain K that minimizes the PI. By using the trace identity

$$\mathrm{tr}(AB) = \mathrm{tr}(BA) \tag{5.3-14}$$

for any compatibly dimensioned matrices A and B (with the trace of a matrix the sum of its diagonal elements), we may write (5.3-11) as

$$J = \tfrac{1}{2}\mathrm{tr}(PX), \tag{5.3-15}$$

where the $n \times n$ symmetric matrix X is defined by

$$X \equiv x(0)x^{\mathrm{T}}(0) \tag{5.3-16}$$

It is now clear that the problem of selecting K to minimize (5.3-8) subject to the dynamical constraint (5.3-7) on the states is equivalent to the *algebraic* problem of selecting K to minimize (5.3-15) subject to the constraint (5.3-13) on the auxiliary matrix P.

To solve this modified problem, we use the Lagrange multiplier approach (Lewis, 1986) to modify the problem yet again. Thus, adjoin the constraint to the PI by defining the Hamiltonian

$$\mathscr{H} = \mathrm{tr}(PX) + \mathrm{tr}(gS) \tag{5.3-17}$$

with S a symmetric $n \times n$ matrix of Lagrange multipliers that still needs to be determined. Then our constrained optimization problem is equivalent to the simpler problem of minimizing (5.3-17) without constraints. To accomplish this we need only set the partial derivatives of $\mathscr{H}$ with respect to all the independent variables P, S, and K equal to zero. Using the facts that for any compatibly dimensioned matrices A, B, and C and any scalar y,

$$\frac{\partial}{\partial B}\mathrm{tr}(ABC) = A^{\mathrm{T}}C^{\mathrm{T}} \tag{5.3-18}$$

and

$$\frac{\partial y}{\partial B^{\mathrm{T}}} = \left[\frac{\partial y}{\partial B}\right]^{\mathrm{T}}, \tag{5.3-19}$$

the necessary conditions for the solution of the LQR problem with output feedback are given by

$$0 = \frac{\partial \mathscr{H}}{\partial S} = g = A_c^{\mathrm{T}}P + PA_c + C^{\mathrm{T}}K^{\mathrm{T}}BKC + Q \tag{5.3-20}$$

$$0 = \frac{\partial \mathscr{H}}{\partial P} = A_cS + SA_c^{\mathrm{T}} + X \tag{5.3-21}$$

$$0 = \frac{1}{2}\frac{\partial \mathscr{H}}{\partial K} = RKCSC^{\mathrm{T}} - B^{\mathrm{T}}PSC^{T} \tag{5.3-22}$$

The first two of these are Lyapunov equations and the third is an equation for the gain K. If R is positive definite (i.e., all eigenvalues greater than zero, which implies nonsingularity; denoted $R > 0$) and CSC^{T} is nonsingular, then (5.3-22) may be solved for K to obtain

$$K = R^{-1}B^{\mathrm{T}}PSC^{\mathrm{T}}(CSC^{\mathrm{T}})^{-1} \tag{5.3-23}$$

To obtain the output feedback gain K minimizing the PI (5.3-4), we need to solve the three coupled equations (5.3-20), (5.3-21), and (5.3-23). This situation is quite strange, for to find K we must determine along the way the values of two auxiliary and apparently unnecessary $n \times n$ matrices, P and S. These auxiliary quantities may, however, not be as unnecessary as it appears, for note that the optimal cost may be determined directly from P and the initial state by using (5.3-11).

The Initial-Condition Problem Unfortunately, the dependence of X in (5.3-16) on the initial state $x(0)$ is undesirable, since it makes the optimal gain dependent on the initial state through Equation (5.3-21). In many applications $x(0)$ may not be known. This dependence is typical of output feedback design. We will see at the end of this chapter that in the case of state feedback it does not occur. Meanwhile, it is usual (Levine and Athans, 1970) to sidestep this problem by minimizing not the PI (5.3-4) but its *expected value*, that is, $E\{J\}$. Then (5.3-11) and (5.3-16) are replaced by

$$E\{J\} = \tfrac{1}{2}E\{x^{\mathrm{T}}(0)Px(0)\} = \tfrac{1}{2}\mathrm{tr}(PX), \tag{5.3-24}$$

where the symmetric $n \times n$ matrix

$$X \equiv E\{x(0)x^{\mathrm{T}}(0)\} \tag{5.3-25}$$

is the initial autocorrelation of the state. It is usual to assume that nothing is known of $x(0)$ except that it is uniformly distributed on a surface described by X. That is, we assume that the actual initial state is unknown but it is nonzero with a certain expected Euclidean norm. For instance, if the initial states are assumed to be uniformly distributed on the unit sphere, then $X = I$, the identity. This is a sensible assumption for the regulator problem, where we are trying to drive arbitrary nonzero initial states to zero.

The design equations for the LQR with output feedback are collected in Table 5.3-1 for convenient reference. We will now discuss their solution for K.

Determining the Optimal Feedback Gain

The importance of this modern LQ approach to control design is that the matrix equations in Table 5.3-1 are used to solve for all the $m \times p$ elements of K at once. This corresponds to closing all the feedback loops simultaneously. Moreover, as long as certain reasonable conditions (to be discussed) on the plant and PI weighting matrices hold, *the closed-loop system is generally guaranteed to be stable*. In view of the trial-and-error successive-loop-closure approach used in stabilizing multivariable systems using classical approaches, this is quite important.

The equations for P, S, and K are coupled nonlinear matrix equations in three unknowns. It is important to discuss some aspects of their solution for the optimal feedback gain matrix K.

Numerical Solution Techniques There are three basic numerical techniques for determining the optimal output feedback gain K. First, we may use a numerical optimization routine such as the Simplex algorithm by Nelder and Mead (1964) and Press et al. (1986). This algorithm would use only (5.3-26) and (5.3-29). For a given value of K, it would solve the Lyapunov equation for P and then use P in the second equation to determine $E\{J\}$. Based on this, it would vary the elements of K to minimize $E\{J\}$. The Lyapunov equation may be solved using, for instance, subroutine ATXPXA (Bartels and Stewart, 1972). See also the NASA controls design package ORACLS (Armstrong, 1980).

TABLE 5.3-1 LQR with Output Feedback

System Model

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

Control

$$u = -Ky$$

Performance Index

$$J = \tfrac{1}{2} E\left[\int_0^\infty \left(x^{\mathrm{T}} Qx + u^{\mathrm{T}} Ru\right) dt\right]$$

with

$$Q \geq 0, \quad R > 0$$

Optimal Gain Design Equations

$$0 = A_c^T P + PA_c + C^{\mathrm{T}} K^{\mathrm{T}} RKC + Q \tag{5.3-26}$$

$$0 = A_c S + SA_c^T + X \tag{5.3-27}$$

$$K = R^{-1} B^{\mathrm{T}} PSC^{\mathrm{T}} (CSC^{\mathrm{T}})^{-1} \tag{5.3-28}$$

where

$$A_c = A - BKC, \quad X = E\{x(0)x^{\mathrm{T}}(0)\}$$

Optimal Cost

$$J = \tfrac{1}{2}\mathrm{tr}(PX) \tag{5.3-29}$$

A second approach for computing K is to use a gradient-based routine such as Davidon-Fletcher-Powell (Press et al., 1986). This routine would use all of the design equations in Table 5.3-1. For a given value of K, it would solve the two Lyapunov equations to find the auxiliary matrices P and S. Then it would use the third design equation in the form (5.3-22). Note that if P satisfies the first Lyapunov equation, then $g = 0$ so that [see (5.3-17)] $E\{J\} = \frac{1}{2}E\{H\}$ and $\partial E\{J\}/\partial K = \frac{1}{2}\partial E\{\mathscr{H}\}/\partial K$. Thus, the third design equation gives the gradient of $E\{J\}$ with respect to K, which would be used by the routine to update the value of K.

Finally, an iterative solution algorithm was presented by Moerder and Calise (1985). It is given in Table 5.3-2. It was shown by Moerder and Calise (1985) that the algorithm converges to a local minimum for J if the following conditions hold.

Conditions for Convergence of the LQ Solution Algorithm

1. There exists a gain K such that A_c is stable. If this is true, we call the system (5.3-1)/(5.3-2) *output stabilizable*.
2. The output matrix C has full row rank p.

TABLE 5.3-2 Optimal Output Feedback Solution Algorithm

1. Initialize:

   Set $k = 0$.

   Determine a gain K_0 so that $A - BK_0C$ is asymptotically stable.

2. kth iteration:

   Set $A_k = A - BK_kC$.

   Solve for P_k and S_k in

   $$0 = A_k^T P_k + P_k A_k + C^{\mathrm{T}} K_k^T R K_k C + Q$$

   $$0 = A_k S_k + S_k A_k^{\mathrm{T}} + X$$

   Set $J_k = \frac{1}{2}\mathrm{tr}(P_k X)$.

   Evaluate the gain update direction

   $$\Delta K = R^{-1} B^{\mathrm{T}} PSC^{\mathrm{T}} (CSC^{\mathrm{T}})^{-1} - K_k$$

   Update the gain by

   $$K_{k+1} = K_k + \alpha \Delta K$$

   where α is chosen so that

   $$A - BK_{k+1}C \text{ is asymptotically stable}$$

   $$J_{k+1} \equiv \frac{1}{2}\mathrm{tr}(P_{k+1} X) \le J_k$$

   If J_{k+1} and J_k are close enough to each other, go to 3.

   Otherwise, set $k = k + 1$ and go to 2.

3. Terminate:

   Set $K = K_{k+1}, J = J_{k+1}$.

   Stop.

3. Control weighting matrix R is positive definite. This means that all the control inputs should be weighted in the PI.
4. Q is positive semidefinite and $(\sqrt{Q}, A)$ is *detectable*. That is, the observability matrix polynomial

   $$O(S) \equiv \begin{bmatrix} sI - A \\ -\sqrt{Q} \end{bmatrix} \tag{5.3-30}$$

   has full rank n for all values of the complex variable s not contained in the left-half plane (Kailath, 1980).

If these conditions hold, the algorithm finds an output feedback gain that stabilizes the plant and minimizes the PI. The detectability condition means that any unstable system modes must be observable in the PI. Then if the PI is bounded, which it is if the optimization algorithm is successful, signals associated with the unstable modes must go to zero as t becomes large, that is, they are stabilized in the closed-loop system.

Initial Stabilizing Gain Since all three algorithms for solving the matrix equations in Table 5.3-1 for K are iterative in nature, a basic issue for all of them is the selection of an initial stabilizing output feedback gain K_0. That is, to start the algorithms, it is necessary to provide a K_0 such that $(A - BK_0C)$ is stable. See, for instance, Table 5.3-2.

One technique for finding such a gain is given by Broussard and Halyo (1983). Another possibility is to use the eigenstructure assignment techniques of the preceding section to determine an initial gain for the LQ solution algorithm. We could even select a stabilizing gain using the classical techniques of Chapter 4 and then use modern design techniques to tune the control gains for optimal performance.

A quite convenient technique for finding an initial stabilizing gain K_0 is discussed in Section 5.5. This involves finding a full $m \times n$ state-variable feedback matrix and then zeroing the entries that are not needed in the $m \times p$ output feedback matrix for the given measured outputs. Note that there are many techniques for finding a full state feedback that stabilizes a system given A and B [see Section 5.7 and Lewis (1986)].

Iterative Design Software that solves for the optimal output feedback gain K is described in Appendix B. Given good software, design using the LQ approach is straightforward. A design procedure would involve selecting the *design parameters* Q and R, determining the optimal gain K, and simulating the closed-loop response and frequency-domain characteristics. If the results are not suitable, different matrices Q and R are chosen and the design is repeated. Good software makes a design iteration take only a few minutes.

This approach introduces the notion of *tuning the design parameters Q and R for good performance*. In the next two sections we will present sensible techniques for obtaining suitable PI weighting matrices Q and R that do not depend on individually selecting all of their entries.

Example 5.3-1 will illustrate these notions.

Selection of the PI Weighting Matrices

Once the PI weighting matrices Q and R have been selected, the determination of the optimal feedback gain K is a formal procedure relying on the solution of nonlinear coupled matrix equations. Therefore, the engineering judgment in modern LQ design appears in the selection of Q and R. There are some guidelines for this which we will now discuss.

Observability in the Choice of Q For stabilizing solutions to the output feedback problem, it is necessary for $(\sqrt{Q}, A)$ to be detectable. The detectability condition basically means that Q should be chosen so that all unstable states are weighted in the PI. Then, if J is bounded so that $\sqrt{Q}x(t)$ vanishes for large t, the open-loop unstable states will be forced to zero through the action of the control. This means exactly that the unstable poles must have been stabilized by the feedback control gain.

A stronger condition than detectability is *observability*, which amounts to the full rank of $O(s)$ for all values of s. Observability is easier to check than detectability since it is equivalent to the full rank n of the *observability matrix*

$$O \equiv \begin{bmatrix} \sqrt{Q} \\ \sqrt{Q}A \\ \vdots \\ \sqrt{Q}A^{n-1} \end{bmatrix}, \tag{5.3-31}$$

which is a constant matrix and so easier to deal with than $O(s)$. In fact, O has full rank n if and only if the observability Gramian $O^{\mathrm{T}}O$ is nonsingular. Since the Gramian is an $n \times n$ matrix, its determinant is easily examined using available software [e.g., singular-value decomposition/condition number (IMSL)]. The observability of $(\sqrt{Q}, A)$ means basically that *all* states are weighted in the PI.

From a numerical point of view, if $(\sqrt{Q}, A)$ is observable, a positive-definite solution P to (5.3-26) results; otherwise, P may be singular. Since P helps determine K through (5.3-28), it is found that if P is singular, it may result in some zero-gain elements in K. That is, if $(\sqrt{Q}, A)$ is not observable, the LQ algorithm can refuse to close some of the feedback loops.

This observability condition amounts to a restriction on the selection of Q and is a drawback of modern control (see Example 5.3-1). In Section 5.5 we will show how to avoid this condition by using a modified PI.

The Structure of Q The choice of Q can be confronted more easily by considering the performance objectives of the LQR. Suppose that a *performance output*

$$z = Hx \tag{5.3-32}$$

is required to be small in the closed-loop system. For instance, in an aircraft lateral regulator it is desired for the sideslip angle, yaw rate, roll angle, and roll rate to be small (see Example 5.3-1). Therefore, we might select $z = [\beta \quad r \quad \phi \quad p]^{\mathrm{T}}$. Once $z(t)$ has been chosen, the performance output matrix H may be formally written down.

The signal $z(t)$ may be made small by LQR design by selecting the PI

$$J = \tfrac{1}{2}\int_0^\infty \left(z^{\mathrm{T}}z + u^{\mathrm{T}}Ru\right)dt, \tag{5.3-33}$$

which amounts to using the PI in Table 5.3-1 with $Q = H^{\mathrm{T}}H$, so that Q may be computed from H. That is, by weighting *performance outputs* in the PI, Q is directly given.

Maximum Desired Values of z(t) and u(t) A convenient guideline for selecting Q and R is given by Bryson and Ho (1975). Suppose that the performance output (5.3-32) has been defined so that H is given. Consider the PI

$$J = \tfrac{1}{2}\int_0^\infty \left(z^{\mathrm{T}}\overline{Q}z + u^{\mathrm{T}}\,Ru\right)dt \tag{5.3-34}$$

Then, in Table 5.3-1 we have $Q = H^{\mathrm{T}}\overline{Q}H$. To select $\overline{Q}$ and R, one might proceed as follows using the *maximum allowable deviations* in $z(t)$ and $u(t)$.

Define the maximum allowable deviation in component $z_i(t)$ of $z(t)$ as z_{iM} and the maximum allowable deviation in component $u_i(t)$ of the control input $u(t)$ as u_{iM}. Then $\overline{Q}$ and R may be selected as $\overline{Q} = \mathrm{diag}\{q_i\}$, $R = \mathrm{diag}\{r_i\}$, with

$$q_i = \frac{1}{z_{iM}^2}, \qquad r_i = \frac{1}{r_{iM}^2} \tag{5.3-35}$$

The rationale for this choice is easy to understand. For instance, as the allowed limits z_{iM} on $z_i(t)$ decrease, the weighting in the PI placed on $z_i(t)$ increases, which requires smaller excursions in $z_i(t)$ in the closed-loop system.

Implicit Model Following The implicit model-following design technique in Section 5.6 shows how to select Q and R so that the closed-loop system behaves like a prescribed ideal model. The ideal model may be selected according to flying qualities requirements (see Section 4.3). It should be selected so that its poles and zeros correspond to the desired closed-loop time-response characteristics.

Asymptotic Properties of the LQR Consider the PI

$$J = \tfrac{1}{2}\int_0^{\infty}(x^{\mathrm{T}}Qx + pu^{\mathrm{T}}Ru)dt, \tag{5.3-36}$$

where ρ is a scalar design parameter. There are some quite nice results that describe the asymptotic performance of the LQR as ρ becomes small and as ρ becomes large (Kwakernaak and Sivan, 1972; Harvey and Stein, 1978; Grimble and Johnson, 1988).

These results detail the asymptotic closed-loop eigenstructure of the LQR and are of some assistance in selecting Q and R. Unfortunately, they are only well developed for the case of full state-variable feedback, where $C = I$ and all the states are allowed for feedback. Thus, they are appropriate in connection with the discussion in Section 5.7.

Example 5.3-1: LQR Design for F-16 Lateral Regulator In Chapter 4 we designed a roll damper/yaw damper for a low-speed flight condition of the F-16. Successive loop closures were used to perform the design using the root-locus approach. In this example we should like to demonstrate the power of the LQ design equations in Table 5.3-1 by designing a lateral regulator.

In our approach we will select the design parameters Q and R in the table and then use the design equations there to close all the feedback loops simultaneously by computing K. The objective is to design a closed-loop controller to provide for the function of a lateral SAS as well as the closure of the roll-attitude loop. This objective involves the design of two feedback channels with multiple loops, but it is straightforward to deal with using modern control techniques. The simplicity of MIMO design using the LQR will be evident.

(a) ***Aircraft State Equations.*** We used the F-16 linearized lateral dynamics at the nominal flight condition in Table 3.6-3 ($V_T = 502\,\text{ft/s}$, 300 psf dynamic pressure, cg at 0.35 $\bar{c}$), retaining the lateral states sideslip, β, bank angle, ϕ, roll rate, p, and yaw rate, r. Additional states δ_a and δ_r are introduced by the aileron and rudder actuators

$$\delta_a = \frac{20.2}{s+20.2}u_a, \quad \delta_r = \frac{20.2}{s+20.2}u_r \tag{1}$$

A washout filter

$$r_w = \frac{s}{s+1}r \tag{2}$$

is used, with r the yaw rate and r_w the washed-out yaw rate. The washout filter state is denoted x_w. Thus, the entire state vector is

$$x = [\beta \quad \phi \quad p \quad r \quad \delta_a \quad \delta_r \quad x_w]^{\mathrm{T}} \tag{3}$$

The full-state-variable model of the aircraft plus actuators, washout filter, and control dynamics is of the form

$$\dot{x} = Ax + Bu, \tag{4}$$

with

$$A = \begin{bmatrix} -0.3220 & 0.0640 & 0.0364 & -0.9917 & 0.0003 & 0.0008 & 0 \\ 0 & 0 & 1 & 0.0037 & 0 & 0 & 0 \\ -30.6492 & 0 & -3.6784 & 0.6646 & -0.7333 & 0.1315 & 0 \\ 8.5396 & 0 & -0.0254 & -0.4764 & -0.0319 & -0.0620 & 0 \\ 0 & 0 & 0 & 0 & -20.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -20.2 & 0 \\ 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20.2 & 0 \\ 0 & 20.2 \\ 0 & 0 \end{bmatrix} \tag{5}$$

The control inputs are the rudder and aileron servo inputs so that

$$u = \begin{bmatrix} u_a \\ u_r \end{bmatrix} \tag{6}$$

and the output is

$$y = \begin{bmatrix} r_w \\ p \\ \beta \\ \phi \end{bmatrix} \tag{7}$$

Thus, $y = Cx$ with

$$C = \begin{bmatrix} 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 \\ 0 & 0 & 57.2958 & 0 & 0 & 0 & 0 \\ 57.2958 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{8}$$

The factor of 57.2958 converts radians to degrees. The feedback control will be output feedback of the form $u = Ky$, so that K is a 2×4 matrix. That is, we will select eight feedback gains.

For this system the open-loop dutch roll mode has poles at $-0.4425 \pm j3.063$ and so has insufficient damping. The spiral mode has a pole at -0.01631.

(b) ***LQR Output Feedback Design.*** For the computation of the feedback gain K, it is necessary to select PI weighting matrices Q and R in Table 5.3-1. Then the software described in Appendix B is used to compute the optimal gain K using the design equations in the table. Our philosophy for selecting Q and R follows.

First, let us discuss the choice of Q. It is desired to obtain good stability of the dutch roll mode, so that β^2 and r^2 should be weighted in the PI by factors of q_{dr}. To obtain stability of the roll mode, which in closed-loop will consist primarily of p and ϕ, we may weight p^2 and ϕ^2 in the PI by factors of q_r. We do not care about δ_a and δ_r, so it is not necessary to weight them in the PI; the control weighting matrix R will prevent unreasonably large control inputs. Thus, so far we have

$$Q = \text{diag}\{q_{dr}, \quad q_r, \quad q_r, \quad q_{dr}, \quad 0, \quad 0, \quad 0\} \tag{9}$$

We do not care directly about x_w; however, it is necessary to weight it in the PI. This is because omitting it would cause problems with the observability condition. A square root of Q in (9) is

$$\sqrt{Q} = \left[\sqrt{q_{dr}} \quad \sqrt{q_r} \quad \sqrt{q_r} \quad \sqrt{q_{dr}} \quad 0 \quad 0 \quad 0\right] \tag{10}$$

Consequently, the observability matrix (5.3-31) has a right-hand column of zero; hence the system is unobservable. This may be noted in simpler fashion by examining the A-matrix in (5), where the seventh state x_w is seen to have no influence on the states that are weighted in (9). To correct this potential problem, we chose

$$Q = \text{diag}\{q_{dr}, q_r, q_r, q_{dr}, 0, 0, 1\} \tag{11}$$

As far as the R-matrix goes, it is generally satisfactory to select it as

$$R = \rho I, \tag{12}$$

with I the identity matrix and ρ a scalar design parameter.

Now the design equations in Table 5.3-1 were solved using the software described in Appendix B for several choices of ρ, q_{dr}, q_r. After a few trials, we obtained a good result using $\rho = 0.1$, $q_{qr} = 50$, $q_r = 100$. For this selection the optimal feedback gain was

$$K = \begin{bmatrix} -0.56 & -0.44 & 0.11 & -0.35 \\ -1.19 & -0.21 & -0.44 & 0.26 \end{bmatrix} \tag{13}$$

The resulting closed-loop poles were at

$$\begin{aligned} s = &- 3.13 \pm j0.83 && \text{dutch roll mode(r, } \beta) \\ &- 0.82 \pm j0.11 && \text{roll mode}(p, \varphi) \\ &-11.47 \pm j17.18, -15.02 && \end{aligned} \tag{14}$$

To verify the design, a simulation was performed. The initial state was selected as $x(0) = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$; that is, we chose $\beta(0) = 1$. Figure 5.3-1 shows the results. Part *a* shows the dutch roll mode and part *b* the roll mode. Note that the responses correspond to the poles in (14), where the dutch roll is the faster mode. Compare to the results of Example 4.4-3.

This design has two deficiencies. First, it uses eight feedback gains in (13). This is undesirable for two reasons: (1) it requires the gain scheduling of all eight gains and (2) the control system has no structure. That is, all outputs are fed back to both inputs; zeroing some of the gains would give the controller more structure in terms of feeding back certain outputs to only one or the other of the inputs.

The second deficiency is that it was necessary to juggle the entries of Q to obtain a good solution. Actually, due to our weighting of β^2 and r^2 by q_{dr} and φ^2 and p^2 by q_r, the design was fairly straightforward and took about half an hour in all. It was, however, necessary to weight the washout filter state x_w, which is not obvious without considering the observability question.

In Section 5.5 we will show how to overcome both of these deficiencies: the former using "constrained output feedback" and the latter using time weighting like t^k in the PI.

(c) ***Effect of Weighting Parameters.*** It is interesting to examine more closely the effects of the design parameters, namely, the entries of the PI weighting matrices Q and R. Using the same Q as above, we show the sideslip response in Figure 5.3-2*a* for control weightings of $\rho = 0.1$, 0.5, and 1. Increased control weighting in the PI generally suppresses the control signals in the closed-loop system; that is, less control effort is allowed. As less control effort is allowed, the control is less effective in controlling the modes. Indeed, according to the figure, as ρ increases, the undershoot in β increases. Moreover, with increasing ρ the control is also less effective in suppressing the undesirable oscillations in the dutch roll mode which were noted in the open-loop system.

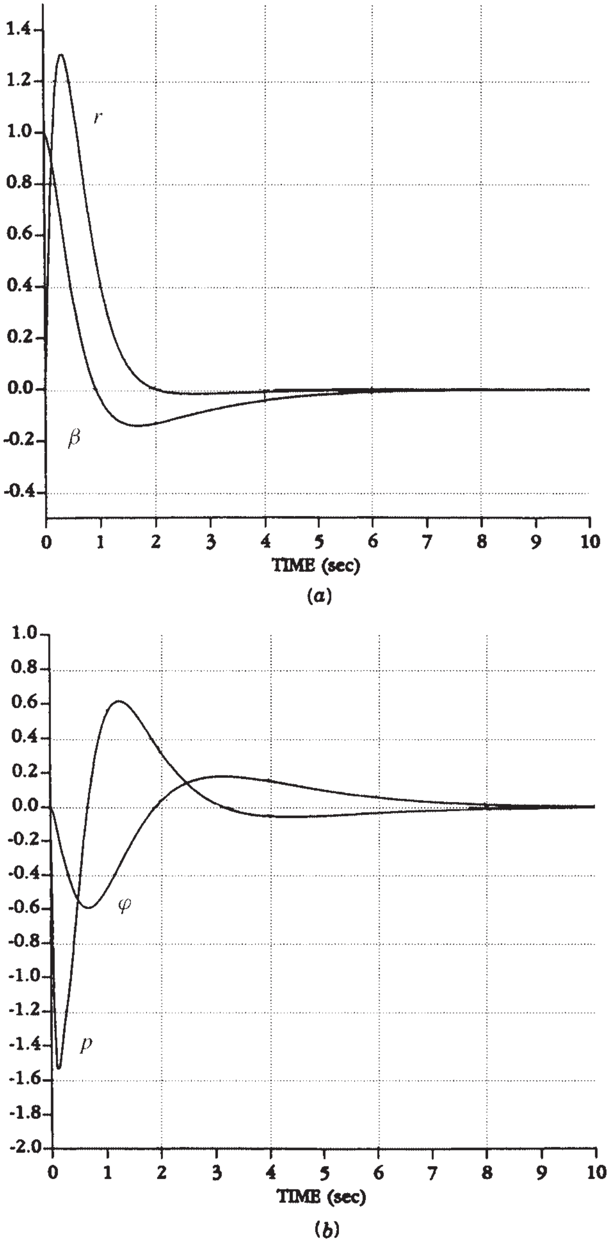


Figure 5.3-1 Closed-loop lateral response: (*a*) dutch roll states β and r; (*b*) roll-mode states ϕ and p.

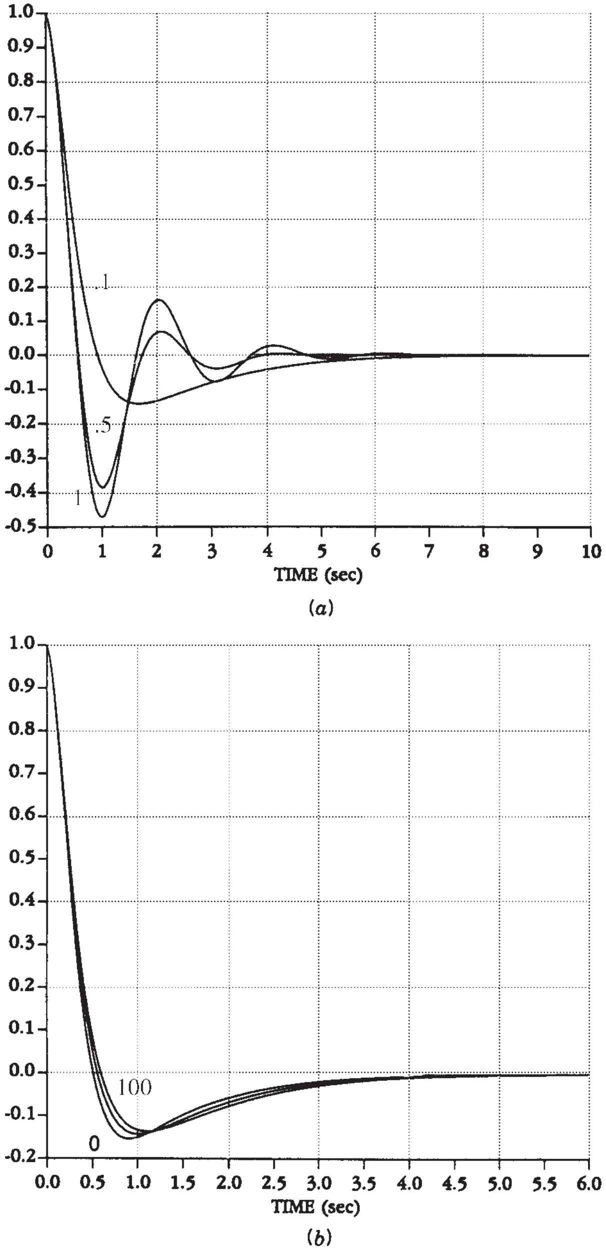


Figure 5.3-2 Effect of PI weighting parameters: (*a*) sideslip as a function of $\rho(\rho = 0.1, 0.5, 1)$; (*b*) sideslip as a function of $q_{\text{dr}}(q_{\text{dr}} = 0.50, 100)$.

As far as the effect of the dutch roll weighting q_{dr} goes, examine Figure 5.3-2*b*, where $\rho = 0.1$ and $q_r = 100$ as in part *a*, but the sideslip response is shown for $q_{dr} = 0$, 50, and 100. As q_{dr} increases, the undershoot decreases, reflecting the fact that increased weighting on β^2 in the PI will result in smaller excursions in β in closed loop.

One last point is worth noting. The open-loop system is stable; therefore, it is clear that it is detectable, since all the unstable modes are observable for any choice of Q (there are no unstable modes). Thus, the design would work if we omitted the weighting on x_w^2 in the Q-matrix (although, it turns out, the closed-loop poles are not as good). In general, however, the detectability condition is difficult to check in large systems that are open-loop unstable; thus, the observability condition is used instead. Failing to weight an undetectable state can lead to some zero elements of K, meaning that some feedback loops are not closed. Thus, to guarantee that this does not occur, Q should be selected so that $(\sqrt{Q}, A)$ is observable.

To avoid all this discussion on observability, we may simply use a modified nonstandard PI with weighting like t^k. Such a PI is introduced in Section 5.5 and leads to a simplified design procedure.

(d) *Gain Scheduling.* For implementation on an aircraft, the control gains in (13) should be gain scheduled. To accomplish this, the nonlinear aircraft equations are linearized at several equilibrium flight conditions over the desired flight envelope to obtain state-variable models like (4) with different A- and B-matrices. Then the LQR design is repeated for those different systems.

A major advantage of LQR design will quickly be apparent, for once the control structure has been selected, it takes only a minute or two to run the software to find the optimal gains for a new A and B using the design equations in Table 5.3-1. Note that the optimal gains for one point in the gain schedule can be used as initial stabilizing gains in the LQ solution algorithm for the next point.

It is important, however, to be aware of an additional consideration. The optimal gains at each gain-scheduling point should guarantee *robust stability and performance*; that is, they should guarantee stability and good performance at points *near* the design equilibrium point. Such robust stability can be verified after the LQ design by using multivariable frequency-domain techniques. These techniques are developed in Section 6.2, where the remarks on robustness to plant parameter variations are particularly relevant to gain scheduling. ■

5.4 TRACKING A COMMAND

In aircraft control we are often interested not in regulating the state near zero, which we discussed in the preceding section, but in *following a nonzero reference command signal*. For example, we may be interested in designing a control system for optimal step-response shaping. This reference input tracking or *servodesign* problem is important in the design of command augmentation systems (CASs), where the reference command may be, for instance, desired pitch rate or normal acceleration. In this section and the next we cover tracker design.

It should be mentioned that the *optimal* LQ tracker of modern control is not a causal system (Lewis, 1986). It depends on solving an "adjoint" system of differential equations backward in time and so is impossible to implement. A suboptimal "steady-state" tracker using full state-variable feedback is available, but it offers no convenient structure for the control system in terms of desired dynamics such as PI control, washout filters, and so on. Thus, there have been problems with using it in aircraft control.

Modified versions of the LQ tracker have been presented by Davison and Ferguson (1981) and Gangsaas et al. (1986). There, controllers of desired structure can be designed since the approaches are output feedback based. The optimal gains are determined numerically to minimize a PI with, possibly, some constraints.

It is possible to design a tracker by first designing a regulator using, for instance, Table 5.3-1. Then some feedforward terms are added to guarantee perfect tracking (Kwakernaak and Sivan, 1972). The problem with this technique is that the resulting tracker has no convenient structure and often requires derivatives of the reference command input. Moreover, servosystems designed using this approach depend on knowing the dc gain exactly. If the dc gain is not known exactly, the performance deteriorates. That is, the design is *not robust* to uncertainties in the model.

Here we discuss an approach to the design of tracking control systems which is more useful in aircraft control applications (Stevens et al., 1992). This approach will allow us to design a servo control system that has any structure desired. This structure will include a unity-gain outer loop that feeds the performance output back and subtracts it from the reference command, thus defining a tracker error $e(t)$ which should be kept small (see Figure 5.4-1). It can also include compensator dynamics such as a washout filter or an integral controller. The control gains are chosen to minimize a quadratic PI. We are able to give explicit design equations for the control gains (see Table 5.4-1), which may be solved using the software described in Appendix B.

A problem with the tracker developed in this section is the need to select the design parameters Q and R in the PI in Table 5.4-1, given in a later subsection. There are

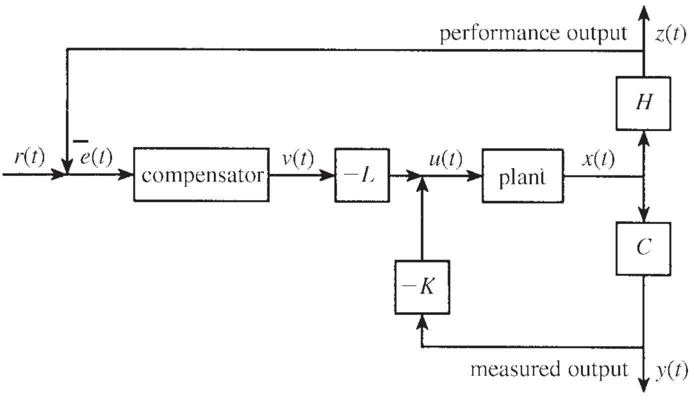


Figure 5.4-1 Plant with compensator of desired structure.

some intuitive techniques available for choosing these parameters (see Section 5.3); however, in Section 5.5 we will show how modified PIs may be used to make the selection of Q and R almost transparent, yielding tracker design techniques that are very convenient for use in aircraft control systems design. We will show, in fact, that *the key to achieving required performance using modern design strategies is in selecting an appropriate PI.*

Tracker with Desired Structure

In aircraft control design there is a wealth of experience and knowledge that dictates in many situations what sort of compensator dynamics yield good performance from the point of view of both the control engineer and the pilot. For example, a washout circuit may be required, or it may be necessary to augment some feedforward channels with integrators to obtain a steady-state error of exactly zero.

The control system structures used in classical aircraft design also give good *robustness properties*. That is, they perform well even if there are disturbances or uncertainties in the system. Thus, the multivariable approach developed here usually affords this robustness. Formal techniques for verifying closed-loop robustness for multivariable control systems are given in Chapter 6.

Our approach to tracker design allows controller dynamics of any desired structure and then determines the control gains that minimize a quadratic PI over that structure. Before discussing the tracker design, let us recall from Chapter 3 how compensator dynamics may be incorporated into the aircraft state equations.

A dynamic compensator of prescribed structure may be incorporated into the system description as follows.

Consider the situation in Figure 5.4-1, where the plant is described by

$$\dot{x} = Ax + Bu \tag{5.4-1}$$

$$y = Cx, \tag{5.4-2}$$

with state $x(t)$, control input $u(t)$, and $y(t)$ the *measured output* available for feedback purposes. In addition,

$$z = Hx \tag{5.4-3}$$

is a *performance output*, which must track the given *reference input* $z(t)$. The performance output $z(t)$ is not generally equal to $y(t)$.

It is important to realize that for perfect tracking it is necessary to have as many control inputs in vector $u(t)$ as there are command signals to track in $r(t)$ (Kwakernaak and Sivan, 1972).

The dynamic compensator has the form

$$\begin{aligned} \dot{w} &= Fw + Ge \\ v &= Dw + Je, \end{aligned} \tag{5.4-4}$$

with state $w(t)$, output $v(t)$, and input equal to the *tracking error*

$$e(t) = r(t) - z(t) \tag{5.4-5}$$

F, G, D, and J are known matrices chosen to include the desired structure in the compensator.

The allowed form for the plant control input is

$$u = -Ky - Lv, \tag{5.4-6}$$

where the constant gain matrices K and L are to be chosen in the control design step to result in satisfactory tracking of $r(t)$. This formulation allows for both feedback and feedforward compensator dynamics.

As we have seen in Chapter 3, these dynamics and output equations may be written in augmented form as

$$\frac{d}{dt}\begin{bmatrix} x \\ w \end{bmatrix} = \begin{bmatrix} A & 0 \\ -GH & F \end{bmatrix}\begin{bmatrix} x \\ w \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ G \end{bmatrix} r \tag{5.4-7}$$

$$\begin{bmatrix} y \\ v \end{bmatrix} = \begin{bmatrix} C & 0 \\ -JH & D \end{bmatrix}\begin{bmatrix} x \\ w \end{bmatrix} + \begin{bmatrix} 0 \\ J \end{bmatrix} r \tag{5.4-8}$$

$$z = [H \;\; 0]\begin{bmatrix} y \\ w \end{bmatrix}, \tag{5.4-9}$$

and the control input may be expressed as

$$u = -[K \quad L]\begin{bmatrix} y \\ v \end{bmatrix} \tag{5.4-10}$$

Note that in terms of the augmented plant/compensator state description, the admissible controls are represented as a *constant output feedback* $[K \quad L]$. In the augmented description, all matrices are known except the gains K and L, which need to be selected to yield acceptable closed-loop performance.

A comment on the compensator matrices F, G, D, and J is in order. Often, these matrices are completely specified by the structure of the compensator. Such is the case, for instance, if the compensator contains integrators. However, if it is desired to include a washout or a lead-lag, it may not be clear exactly how to select the time constants. In such cases, engineering judgment will usually give some insight. However, it may sometimes be necessary to go through the design to be proposed and then, if required, return to readjust F, G, D, and J and reperform the design.

LQ Formulation of the Tracker Problem

By redefining the state, the output, and the matrix variables to streamline the notation, we see that the augmented equations (5.4-7) to (5.4-9) that contain the dynamics of both the aircraft and the compensator are of the form

$$\dot{x} = Ax + Bu + Gr \tag{5.4-11}$$

$$y = Cx + Fr \tag{5.4-12}$$

$$z = Hx \tag{5.4-13}$$

In this description, let us take the state $x(t) \in \mathbf{R}^n$, control input $u(t) \in \mathbf{R}^m$, reference input $r(t) \in \mathbf{R}^q$, performance output $z(t) \in \mathbf{R}^q$, and measured output $y(t) \in \mathbf{R}^p$. The admissible controls (5.4-10) are proportional output feedbacks of the form

$$u = -Ky = -KCx - KFr \tag{5.4-14}$$

with constant gain K to be determined. This situation corresponds to the block diagram in Figure 5.4-2. Since K is an $m \times p$ matrix, we intend to close all the feedback loops simultaneously by computing K.

Using these equations the closed-loop system is found to be

$$\begin{aligned} \dot{x} &= (A - BKC)x + (G - BKF)r \\ &\equiv A_c x + B_c r \end{aligned} \tag{5.4-15}$$

In the remainder of this subsection, we will use the formulation (5.4-11) to (5.4-14), assuming that the compensator, if required, has already been included in the system dynamics and demonstrating how to select the constant output feedback gain matrix K using LQ techniques.

Our formulation differs sharply from the traditional formulations of the optimal tracker problem (Kwakernaak and Sivan, 1972; Lewis, 1986). Note that (5.4-14) includes both feedback and feedforward terms, so that both the closed-loop poles *and compensator zeros* may be affected by varying the gain K (see Example 5.4-1). Thus, we should expect better success in shaping the step response than by placing only the poles.

Since the performance specifications of aircraft are often given in terms of time-domain criteria (*Mil. Spec. 1797*, 1987) (see Section 4.3) and these criteria are

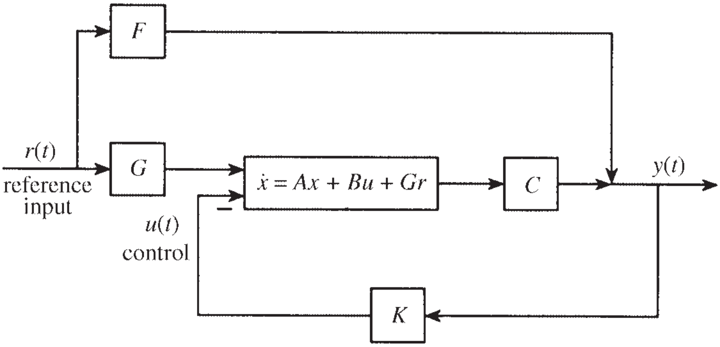


Figure 5.4-2 Plant/feedback structure.

closely related to the step response, we will assume henceforth that the reference input $r(t)$ is a step command with magnitude r_0. Designing for such a command will yield suitable time-response characteristics. Although our design is based on step-response shaping, it should be clearly realized that the resulting control system, if properly designed, will give good time responses for *any arbitrary reference command signal* $r(t)$.

Let us now formulate an optimal control problem for selecting the control gain K to guarantee tracking of $r(t)$. Then we will derive the design equations in Table 5.4-1, which are used to determine the optimal K. These equations are solved using software like that described in Appendix B.

The Deviation System Denote steady-state values by overbars and deviations from the steady-state values by tildes. Then the state, output, and control deviations are given by

$$\tilde{x}(t) = x(t) - \bar{x} \tag{5.4-16}$$

$$\tilde{y}(t) = y(t) - \bar{y} = C\tilde{x} \tag{5.4-17}$$

$$\tilde{z}(t) = z(t) - \bar{z} = H\tilde{x} \tag{5.4-18}$$

$$\tilde{u}(t) = u(t) - \bar{u} = -KCx - KFr_0 - (-KC\tilde{x} - KFr_0) = -KC\tilde{x}(t)$$

or

$$\tilde{u} = -K\tilde{y} \tag{5.4-19}$$

The tracking error $e(t) = r(t) - z(t)$ is given by

$$e(t) = \tilde{e}(t) + \bar{e} \tag{5.4-20}$$

with the error deviation given by

$$\tilde{e}(t) = e(t) - \bar{e} = (r_0 - Hx) - (r_0 - H\tilde{x}) = -H\tilde{x}$$

or

$$\tilde{e} = -\tilde{z} \tag{5.4-21}$$

Since in any acceptable design the closed-loop plant will be asymptotically stable, A_c is nonsingular. According to (5.4-15), at steady state

$$0 = A_c\bar{x} + B_x r_0, \tag{5.4-22}$$

so that the steady-state response $\bar{x}$ is

$$x = -A_c^{-1} B_c r_0 \tag{5.4-23}$$

and the steady-state error is

$$\overline{e} = r_0 - H\overline{x} = (1 + HA_c^{-1}B_c)r_0 \tag{5.4-24}$$

To understand this expression, note that the closed-loop transfer function from r_0 to z [see (5.4-15) and (5.4-13)] is

$$H(s) = H(sI - A_c)^{-1}B_c \tag{5.4-25}$$

The steady-state behavior may be investigated by considering the dc value of $H(s)$ (i.e., $s_{=0}$); this is just $-HA_c^{-1}B_c$, the term appearing in (5.4-24).

Using (5.4-16), (5.4-19), and (5.4-23) in (5.4-15) the closed-loop dynamics of the state deviation are seen to be

$$\dot{\widetilde{x}} = A_c\widetilde{x} \tag{5.4-26}$$

$$\widetilde{y} = C\widetilde{x} \tag{5.4-27}$$

$$\widetilde{z} = H\widetilde{x} = -\widetilde{e} \tag{5.4-28}$$

and the control input to the deviation system (5.4-26) is (5.4-19). Thus, the step-response shaping problem has been converted to a *regulator problem* for the deviation system

$$\dot{\widetilde{x}} = A\widetilde{x} + B\widetilde{u} \tag{5.4-29}$$

Again, we emphasize the difference between our approach and traditional ones (e.g., Kwakernaak and Sivan, 1972). Once the gain K in (5.4-19) has been found, the control for the plant is given by (5.4-14), which inherently has both feedback and feedforward terms. Thus, no extra feedforward term need be added to make $\overline{e}$ zero.

Performance Index To make the tracking error $e(t)$ in (5.4-20) small, we propose to attack two equivalent problems: the problem of regulating the error deviation $\widetilde{e}(t) = -\widetilde{z}(t)$ to zero and the problem of making the steady-state error $\overline{e}$ small.

Note that we do not assume a type-1 system, which would force $\overline{e}$ to be equal to zero. This can be important in aircraft control, where it may not be desirable to force the system to be of type 1 by augmenting all control channels with integrators. This augmentation complicates the servo structure. Moreover, it is well known from classical control theory that suitable step responses may often be obtained without resorting to inserting integrators in all the feedforward channels.

To make both the error deviation $\overline{e}(t) = -H\widetilde{x}(t)$ and the steady-state error $\overline{e}$ small, we propose selecting K to minimize the PI

$$J = \tfrac{1}{2}\int_0^\infty \left(\widetilde{e}^{\mathrm{T}}\widetilde{e} + \widetilde{u}^{\mathrm{T}}R\widetilde{u}\right)\, dt + \tfrac{1}{2}\overline{e}^{\mathrm{T}}V\overline{e}, \tag{5.4-30}$$

with $R > 0$, $V \geq 0$, design parameters. The integrand is the standard quadratic PI with, however, a weighting V included on the steady-state error. Note that the PI weights the control *deviations* and not the controls themselves. If the system is of type 1, containing integrators in all the feedforward paths, then V may be set to zero since the steady-state error is automatically zero.

Making the error deviation $\tilde{e}(t)$ small improves the transient response, while making the steady-state error $\bar{e}(t)$ small improves the steady-state response. If the system is of type 0, these effects involve a trade-off, so that then there is a design trade-off involved in selecting the size of V.

We can generally select $R = rI$ and $V = vI$, with r and v scalars. This simplifies the design since now only a few parameters must be tuned during the interactive design process.

According to (5.4-21), $\tilde{e}^{\mathrm{T}}\tilde{e} = \tilde{x}^{\mathrm{T}}H^{\mathrm{T}}H\tilde{x}$. Referring to Table 5.3-1, therefore, it follows that the matrix Q there is equal to $H^{\mathrm{T}}H$, where H is known. That is, weighting the error deviation in the PI has already shown us how to select the design parameter Q, affording a considerable simplification.

The problem we now have to solve is how to select the control gains K to minimize the PI J for the deviation system (5.4-29). Then the tracker control for the original system is given by (5.4-14).

We should point out that the proposed approach is suboptimal in the sense that minimizing the PI does not necessarily minimize a quadratic function of the total error $e(t) = \bar{e} + \tilde{e}(t)$. It does, however, guarantee that both $\tilde{e}(t)$ and $\bar{e}$ are small in the closed-loop system, which is a design goal.

Solution of the LQ Tracker Problem

It is now necessary to solve for the optimal feedback gain K that minimizes the PI. The design equations needed are now derived. They appear in Table 5.4-1.

By using (5.4-26) and a technique like the one used in Section 5.3 (see the problems at the end of the chapter), the optimal cost is found to satisfy

$$J = \tfrac{1}{2}\tilde{x}^{\mathrm{T}}(0)P\tilde{x}(0) + \tfrac{1}{2}\bar{e}^{\mathrm{T}}V\bar{e}, \tag{5.4-31}$$

with $P \geq 0$ the solution to

$$0 = g \equiv A_c^{\mathrm{T}}P + PA_c + Q + C^{\mathrm{T}}K^{\mathrm{T}}RKC, \tag{5.4-32}$$

with $Q = H^{\mathrm{T}}H$ and $\bar{e}$ given by (5.4-24).

In our discussion of the linear quadratic regulator we assumed that the initial conditions were uniformly distributed on a surface with known characteristics. While this is satisfactory for the regulator problem, it is an unsatisfactory assumption for the tracker problem. In the latter situation the system starts at rest and must achieve a given final state that is dependent on the reference input, namely (5.4-23). To find the correct value of $\bar{x}(0)$, we note that since the plant starts at rest [i.e., $x(0) = 0$], according to (5.4-16),

$$\tilde{x}(0) = -\bar{x}, \tag{5.4-33}$$

so that the optimal cost (5.4-31) becomes

$$J = \tfrac{1}{2}\overline{x}^{\mathrm{T}}P\overline{x} + \tfrac{1}{2}\overline{e}^{\mathrm{T}}V\overline{e} = \tfrac{1}{2}\mathrm{tr}(PX) + \tfrac{1}{2}\overline{e}^{\mathrm{T}}V\overline{e}, \tag{5.4-34}$$

with P given by (5.4-32), $\overline{e}$ given by (5.4-24), and

$$X \equiv \overline{x}\,\overline{x}^{\mathrm{T}} = A_c^{-1}B_c r_0 r_0{}^{\mathrm{T}} B_c^{\mathrm{T}} A_c^{-\mathrm{T}}, \tag{5.4-35}$$

with $A_c^{-\mathrm{T}} = (A_c^{-1})^{\mathrm{T}}$.

The optimal solution to the unit-step tracking problem, with (5.4-11) initially at rest, may now be determined by minimizing J in (5.4-34) over the gains K, subject to the constraint (5.4-32) and Equations (5.4-24) and (5.4-35).

This algebraic optimization problem can be solved by any well-known numerical method (see Press et al., 1986; Söderström, 1978). A good approach for a fairly small number ($mp \leq 10$) of gain elements in K is the Simplex minimization routine (Nelder and Mead, 1964). To evaluate the PI for each fixed value of K in the iterative solution procedure, one may solve (5.4-32) for P using subroutine ATXPXA (Bartels and Stewart, 1972) and then employ (5.4-34). Software for determining the optimal control gains K is described in Appendix B.

Design Equations for Gradient-Based Solution As an alternative solution procedure, one may use gradient-based techniques [e.g., the Davidon-Fletcher-Powell algorithm (Press et al., 1986)], which are generally faster than non-gradient-based approaches.

To find the gradient of the PI with respect to the gains, define the Hamiltonian

$$\mathscr{H} = \tfrac{1}{2}\mathrm{tr}(PX) + \tfrac{1}{2}\mathrm{tr}(gS) + \tfrac{1}{2}\overline{e}^{\mathrm{T}}V\overline{e}, \tag{5.4-36}$$

with S a Lagrange multiplier. Now, using the basic matrix calculus identities,

$$\frac{\partial Y^{-1}}{\partial x} = -Y^{-1}\frac{\partial Y}{\partial x}Y^{-1} \tag{5.4-37}$$

$$\frac{\partial UV}{\partial x} = \frac{\partial U}{\partial x}V + U\frac{\partial V}{\partial x} \tag{5.4-38}$$

$$\frac{\partial y}{\partial x} = \mathrm{tr}\left[\frac{\partial y}{\partial z}\cdot\frac{\partial z^{\mathrm{T}}}{\partial x}\right], \tag{5.4-39}$$

we may proceed as in the preceding section, with, however, a little more patience due to the extra terms (see the problems!), to obtain the necessary conditions for a solution given in Table 5.4-1.

To find K by a gradient minimization algorithm, it is necessary to provide the algorithm with the values of J and $\partial J/\partial K$ for a given K. The value of J is given by the expression in Table 5.4-1 for the optimal cost. To find $\partial J/\partial K$ given K, solve (5.4-40) and (5.4-41) for P and S. Then since these equations hold, $\partial J/\partial K = \partial\mathscr{H}/\partial K$, which may be found using (5.4-42). These equations should be compared to those in Table 5.3-1. Note that the dependence of X on the gain K [see (5.4-45)] and the presence of $\overline{e}$ in the PI have resulted in extra terms being added in (5.4-42).

TABLE 5.4-1 LQ Tracker with Output Feedback

System Model

$$\dot{x} = Ax + Bu + Gr$$

$$y = Cx + Fr$$

$$z = Hx$$

Control

$$u = -Ky$$

Performance Index

$$J = \tfrac{1}{2}\int_0^\infty (\tilde{x}^{\mathrm{T}} Q\tilde{x} + \tilde{u}^{\mathrm{T}} R\tilde{u})dt + \tfrac{1}{2}\overline{e}^{\mathrm{T}} V\overline{e}, \text{ with } Q = H^{\mathrm{T}}H$$

Optimal Output Feedback Gain

$$0 = \frac{\partial \mathscr{H}}{\partial S} = A_c^{\mathrm{T}} P + PA_c + Q + C^{\mathrm{T}} K^{\mathrm{T}} RKC \tag{5.4-40}$$

$$0 = \frac{\partial \mathscr{H}}{\partial P} = A_c S + SA_c^{\mathrm{T}} + X \tag{5.4-41}$$

$$0 = \frac{\partial \mathscr{H}}{\partial K} = RKCSC^{\mathrm{T}} - B^{\mathrm{T}} PSC^{\mathrm{T}} + B^{\mathrm{T}} A_C^{-\mathrm{T}} (P + H^{\mathrm{T}} VH)\overline{xy}^{\mathrm{T}}$$
$$- B^{\mathrm{T}} A_C^{-\mathrm{T}} H^{\mathrm{T}} V r_0 \overline{y}^{\mathrm{T}}. \tag{5.4-42}$$

with r a unit step of magnitude r_0 and

$$\overline{x} = -A_c^{-1} B_c r_0 \tag{5.4-43}$$

$$\overline{y} = C\overline{x} + Fr_0 \tag{5.4-44}$$

$$X = \overline{xx}^{\mathrm{T}} = A_c^{-1} B_c r_0 r_0{}^{\mathrm{T}} B_c^{\mathrm{T}} A_c^{-\mathrm{T}} \tag{5.4-45}$$

where

$$A_c = A - BKC, \qquad B_c = G - BKF$$

Optimal Cost

$$J = \tfrac{1}{2}\mathrm{tr}(PX) + \tfrac{1}{2}\overline{e}^{\mathrm{T}} V\overline{e}$$

Determining the Optimal Feedback Gain

The issues in finding the optimal output feedback gain K in the tracker problem of Table 5.4-1 are the same as those discussed in connection with the regulator problem of Table 5.3-1. They are choice of Q to satisfy detectability, choice of solution technique, finding an initial stabilizing gain, and iterative design by tuning Q and R.

We emphasize that there are only a few design parameters in our approach, namely, r and v (since we can can generally select $R = rI$, $V = vI$). Thus, it is not difficult or time consuming to come up with good designs. Much of the simplicity of our approach derives from the fact that ^{-}Q in the PI is equal to $H^{\mathrm{T}}H$, which is known.

Let us now illustrate the servo design procedure by an example.

Example 5.4-1: Normal Acceleration CAS In Chapter 4 we designed a normal acceleration CAS using classical control theory. In that example, successive loop closures were used with root-locus design to obtain the feedback gains. Here we will show that using the LQ design equations in Table 5.4-1 we can close all the loops simultaneously. Thus, the design procedure is more straightforward. We will also demonstrate that using LQ design *the algorithm automatically selects the zero of the compensator for optimal performance*.

(a) *Control System Structure.* The normal acceleration control system is shown in Figure 5.4-3, where r is a reference step input in gs and $u(t)$ is the elevator actuator voltage. An integrator has been added in the feedforward path to achieve zero steady-state error. The performance output that should track the reference command r is $z = n_z$, so that the tracking error is $e = r - n_z$. The state and measured outputs are

$$x = \begin{bmatrix} \alpha \\ q \\ \delta_e \\ \alpha_F \\ \epsilon \end{bmatrix}, \qquad y = \begin{bmatrix} \alpha_F \\ q \\ e \\ \epsilon \end{bmatrix}, \tag{1}$$

with $\epsilon(t)$ the integrator output and α_F the filtered measurements of angle of attack.

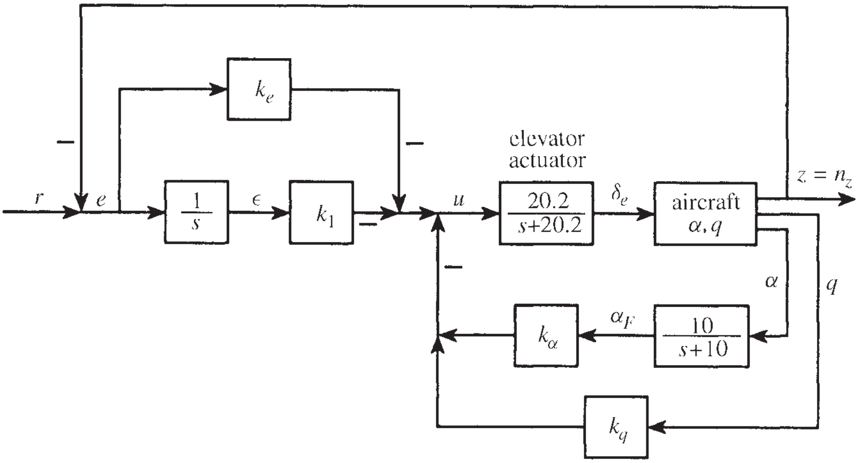


Figure 5.4-3 G-command system.

Exactly as in Chapter 4, we linearized the F-16 dynamics about the nominal flight condition in Table 3.6-3 (502 ft/s, level flight, dynamic pressure of 300 psf, $x_{cg} = 0.35\bar{c}$) and augmented the dynamics to include the elevator actuator, angle-of-attack filter, and compensator dynamics. The result is

$$\dot{x} = Ax + Bu + Gr \tag{2}$$

$$y = Cx + Fr \tag{3}$$

$$z = Hx, \tag{4}$$

with

$$A = \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 & 0 & 0 \\ 0.8225 & -1.07741 & -0.17555 & 0 & 0 \\ 0 & 0 & -20.2 & 0 & 0 \\ 10 & 0 & 0 & -10 & 0 \\ -16.26 & -0.9788 & 0.04852 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 20.2 \\ 0 \\ 0 \end{bmatrix} \qquad G = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{5a}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 57.2988 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 \\ -16.26 & -0.9788 & 0.04852 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad F = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{5b}$$

$$H = [16.26 \quad 0.9788 \quad 0.04852 \quad 0 \quad 0] \tag{5c}$$

The factor of 57.2958 is added to convert angles from radians to degrees.

The control input is

$$u = -Ky = -[k_\alpha \quad k_q \quad k_e \quad k_I]y = -k_\alpha \alpha_F - k_q q - k_e e - k_I \epsilon \tag{6}$$

It is desired to select the four control gains to guarantee a good response to a step command r. Note that k_α and k_q are feedback gains, while k_e and k_I are feedforward gains.

Note that the proportional-plus-integral compensator is given by

$$k_e + \frac{k_I}{s} = k_e \frac{s + k_I/k_e}{s}, \tag{7}$$

which has a zero at $s = -k_I/k_e$. Since the LQ design algorithm will select all four control gains, it will *automatically select the optimal location for the compensator zero.*

(b) *Performance Index and Determination of the Control Gains.* Due to the integrator, the system is of type 1. Therefore, the steady-state error $\overline{e}$ is automatically equal to zero. A natural PI thus seems to be

$$J = \tfrac{1}{2}\int_0^\infty \left(\widetilde{e}^2 + \rho\widetilde{u}^2\right) dt \tag{8}$$

with ρ a scalar weighting parameter. Since $\widetilde{e} = H\widetilde{x}$, this corresponds to the PI in Table 5.4-1 with

$$Q = H^{\mathrm{T}}H = \begin{bmatrix} 264 & 16 & 1 & 0 & 0 \\ 16 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{9}$$

This is, unfortunately, not a suitable Q-matrix since (H,A) is not observable in open loop. Indeed, according to Figure 5.4-3, observing the first two states α and q can never give information about ϵ in the open-loop configuration (where the control gains are zero). Thus, the integrator state is unobservable in the PI. Since the integrator pole is at $s = 0$, (H, A) is undetectable (unstable unobservable pole), so that any design based on (9) would, in fact, yield a value for the integral gain of $k_I = 0$.

We will show in Section 5.5 a very convenient way to correct problems like this. There we will introduce a time weighting of t^k into the PI. In the meantime, to correct the observability problem here, let us select

$$Q = \begin{bmatrix} 264 & 16 & 1 & 0 & 0 \\ 16 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{10}$$

where we include a weighting on $\epsilon(t)$ to make it observable in the PI.

Now, we selected $\rho = 1$ and solved the design equations in Table 5.4-1 for the optimal control gain K using the software described in Appendix B. For this Q and ρ the feedback matrix was

$$K = \begin{bmatrix} 0.006 & -0.152 & 1.17 & 0.996 \end{bmatrix} \tag{11}$$

and the closed-loop poles were

$$\begin{aligned} s &= -1.15 \pm j0.69 \\ &\quad -1.60, -9.98, -19.54 \end{aligned} \tag{12}$$

These yield a system that is not fast enough; the complex pair is also unsuitable in terms of flying qualities requirements.

After repeating the design using several different Q and ρ, we decided on

$$Q = \begin{bmatrix} 264 & 16 & 1 & 0 & 0 \\ 16 & 60 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 \end{bmatrix}, \tag{13}$$

$\rho = 0.01$. The decreased control weighting ρ has the effect of allowing larger control effort and so speeding up the response. The increased weighting on the integrator output $\epsilon(t)$ has the effect of forcing n_z to its final value of r more quickly, hence also speeding up the response. The increased weighting on the second state component q has the effect of regulating excursions in $\tilde{q}(t)$ closer to zero and hence of providing increased damping.

With this Q and ρ the control matrix was

$$K = [-1.629 \quad -1.316 \quad 18.56 \quad 77.6] \tag{14}$$

and the closed-loop poles were at

$$\begin{aligned} s = &-2.98 \pm j3.17, \\ &-19.31 \pm j4.64 \\ &-5.91 \end{aligned} \tag{15}$$

The closed-loop step response is shown in Figure 5.4-4; it is fairly fast with an overshoot of 6%. Note the hump in the initial response due to the non-minimum-phase zero. Further tuning of the elements of Q and R could provide less overshoot, a faster response, and a smaller gain for the angle-of-attack feedback. (It is worth noting that we will obtain a far better response with more reasonable gains in Example 5.5-2, where we use a PI with time-dependent weighting like t^k.)

According to (7), the compensator zero has been placed by the LQ algorithm at

$$s = -\frac{k_1}{k_e} = -4.18 \tag{16}$$

Using the software described in Appendix B, the entire design, including determining K for different choices of Q and ρ until a suitable design was reached, took about 30 minutes.

(c) *Discussion.* We can now emphasize an important aspect of modern LQ design. As long as $Q \geq 0,\ R > 0$, and $(\sqrt{Q},\ A)$ is observable, the closed-loop system designed using Table 5.4-1 is generally stable. Thus, the LQ theory has allowed us to tie the control system design to some *design parameters* which may be tuned to obtain acceptable behavior—namely, the elements of weighting matrices Q and R. Using the software described in Appendix B, for a given Q and R the optimal

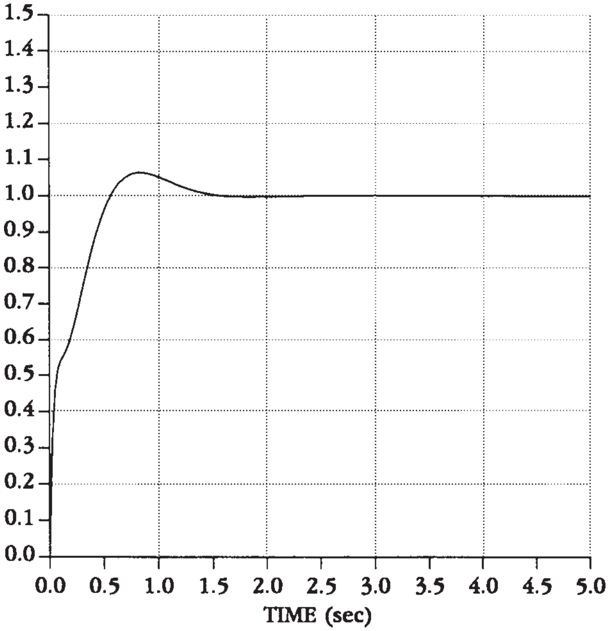


Figure 5.4-4 Normal acceleration step response.

gain K is easily found. If it is not suitable in terms of time responses and closed-loop poles, the elements of Q and R can be changed and the design repeated. The importance of this is that for admissible Q and R *closed-loop stability is guaranteed.*

A disadvantage of the design equations in Table 5.4-1 is the need to try different Q and R until suitable performance is obtained as well as the need for (H, A) to be observable. In Section 5.5 we will introduce a different PI with time weighting of t^k which eliminates these deficiencies.

Another point needs to be made. Using the control (6)/(3) in (2) yields the closed-loop plant

$$\dot{x} = (A - BKC)x + (G - BKF)r \tag{17}$$

whence the closed-loop transfer function from $r(t)$ to $z(t)$ is

$$H(s) = H(sI - (A - BKC))^{-1}(G - BKF) \tag{18}$$

Note that the transfer function numerator depends on the optimal gain K. That is, this scheme uses optimal positioning of *both the poles and zeros* to attain step-response shaping.

(d) *Selection of Initial Stabilizing Gain.* In order to initialize the algorithm that determines the optimal K by solving the design equations in Table 5.4-1, it is necessary to find an initial gain that stabilizes the system. In this example we simply selected gains with signs corresponding to the static loop sensitivity of the individual transfer functions, since this corresponds to negative feedback. The static loop sensitivities from u to α and from u to q are negative, so negative gains were chosen for these loops. The initial gain used was

$$K = [-0.1 \quad -0.1 \quad 1 \quad 1] \tag{19}$$

■

5.5 MODIFYING THE PERFORMANCE INDEX

Modern control theory affords us the ability to close all the feedback loops simultaneously by solving matrix equations for the gain matrix K. With a sensible problem formulation, it also *guarantees the stability of the closed-loop system*. These two fundamental properties make modern design very useful for aircraft control systems. One should recall the difficulty in guaranteeing closed-loop stability in multiloop control systems using one-loop-at-a-time design (Chapter 4).

An additional important advantage is as follows. The standard aircraft control system structures used in classical design have been developed to yield good *robustness properties*. That is, they yield good performance even if there are disturbances in the systems or modeling inaccuracies such as plant parameter variations or high-frequency unmodeled dynamics (e.g., flexible aircraft modes). Since the approach described here allows these standard structures to be incorporated into the control system, it generally yields robust control systems. We will discuss procedures for formally verifying robustness in Chapter 6.

In the LQ regulator design method of Section 5.3 and the LQ tracker design method of Section 5.4, it was necessary to select the PI weighting matrices Q and R as design parameters. Moreover, it was necessary to satisfy an observability property in selecting Q. There are some good approaches that give guidance in selecting Q, such as Bryson's approach (see Section 5.3). Note also that, in Table 5.4-1, $Q=H^TH$, where H is known. However, due to the observability requirement the design parameters Q and R do not necessarily correspond to actual performance objectives.

In this section we show how to modify the PI to considerably simplify the selection of the weighting matrices Q and R in Table 5.4-1. The observability of $(\sqrt{Q},\ A)$ will be unnecessary. The PIs shown in this section correspond to actual performance objectives and involve only a few design parameters, even for systems with many states and many control gains to determine. These facts, coupled with the capability already demonstrated of employing a compensator with any desired structure, will result in a powerful and convenient approach to the design of multivariable aircraft control systems.

A wide range of performance objectives may be attained by using modifications of the PI. We will consider several modifications, all of which are useful depending on the performance objectives. The important concept to grasp is that *the key to obtaining suitable closed-loop behavior using LQ design lies in selecting a suitable PI for the problem at hand*. At the end of the section we present several examples in aircraft control design to demonstrate this issue as well as the directness of the approach.

We will again be concerned with the system plus compensator

$$\dot{x} = Ax + Bu + Gr \tag{5.5-1}$$

We are trying to determine controls that are static output feedbacks of the form

$$u = -Ky \tag{5.5-2}$$

with

$$y = Cx + Fr \tag{5.5-3}$$

the measured output and

$$z = Hx \tag{5.5-4}$$

the performance output, which is to track the reference command r. If we are interested in regulation and not tracking, then G and F do not appear in the equations and z is not defined.

Constrained Feedback Matrix

In many applications it is desired for certain elements of the feedback gain matrix K to be zero to avoid coupling between certain output-input pairs. Zeroing certain gains allows us to specify the detailed structure of the control system. For instance, it may be desired that the error in channel 1 of the controller not be coupled to the control input in channel 2. Zeroing some gains also simplifies the gain-scheduling problem by reducing the number of nonzero gains requiring tabulation. This is called *constrained* output feedback design.

Gain Element Weighting Certain elements k_{ij} of K can be made small simply by weighting them in the performance index, that is, by selecting a PI like

$$J = \frac{1}{2}\int_0^\infty \left(\tilde{x}^{\mathrm{T}}\ Q\tilde{x} + \tilde{u}^{\mathrm{T}}R\tilde{u}\right)\ dt + \sum_i \sum_j g_{ij}k_{ij}^2 \tag{5.5-5}$$

Gain element weight g_{ij} is chosen large to make the (i, j)th element k_{ij} of the feedback matrix K small in the final design. Then, in implementing the controller, the small elements of K may simply be set to zero.

The design problem is now to minimize

$$J = \frac{1}{2}\mathrm{tr}(PX) + \sum_i \sum_j g_{ij}k_{ij}^2, \tag{5.5-6}$$

with P satisfying the matrix equation in Table 5.3-1 or Table 5.4-1, as appropriate. This may be accomplished by using the equations in Table 5.3-1 (if we are interested in regulation) or Table 5.4-1 (if we are interested in tracking) to numerically minimize the PI, but with the extra term involving the gain weighting that appears in (5.5-6) (Moerder and Calise, 1985).

Computing an Initial Stabilizing Gain The iterative algorithms that solve the design equations in Tables 5.3-1 and 5.4-1 require initial stabilizing gains. Unfortunately, stabilizing output feedback gains can be complicated to find in large multivariable systems. A few ways to find K_0 so that $(A-BK_0C)$ is stable were mentioned in Section 5.3 and Example 5.4-1*d*. Gain element weighting can be used to solve the problem of determining an initial stabilizing output feedback gain, as we now see.

There are many techniques for finding a stabilizing *state-variable* feedback given the plant system matrix A and control matrix B [see Kailath (1980) as well as Section 5.7]. That is, it is straightforward to find a K_0 so that $(A-BK_0)$ is stable. Routines that perform this are available in standard software packages such as ORACLS (Armstrong, 1980). Unfortunately, for flight control purposes, state feedback design is unsuitable for reasons such as those we have discussed. However, suppose that an $m \times n$ stabilizing state feedback gain has been found. Then, to determine an $m \times p$ output feedback gain, it is only necessary to weight in the PI the elements of the state feedback matrix that do not correspond to measured outputs. The algorithm will then provide a suitable output feedback gain matrix by driving these elements to zero.

Gain Element Fixing There is an attractive alternative to gain element weighting for fixing gain matrix elements. If a numerical technique such as Simplex (Press et al., 1986) is used to determine the optimal control by varying K and directly evaluating J, we may simply fix certain elements of K and not allow the Simplex to vary them. This allows the fixed elements to be retained at any desired (possibly nonzero) value and takes far fewer computations than gain element weighting, especially if many elements of K are fixed.

If, on the other hand, a gradient-based routine such as Davidon-Fletcher-Powell (Press et al., 1986) is used in conjunction with the design equations in Tables 5.3-1 or 5.4-1, it is easy to modify the gradient $\partial J/\partial K$ to leave certain elements of K fixed. Indeed, to fix element (i, j) of K, one need only set element (i, j) of $\partial J/\partial K$ equal to zero.

These approaches require fewer operations than the gain weighting approach based on (5.5-5) and are incorporated in the software described in Appendix B, which is called program LQ. Illustrations of control design using constrained output feedback are provided in the examples.

Derivative Weighting

As we will soon show in an example, it is often convenient to weight in the PI not the states themselves but their derivatives. This is because rates of change of the states can in some design specifications be more important than the values of the states. For instance, elevator rate of change has a closer connection with required control energy than does elevator deflection. To accomodate such situations, we may consider the PI

$$J = \tfrac{1}{2}\int_0^\infty \dot{x}^{\mathrm{T}} Q \dot{\tilde{x}}\ dt \tag{5.5-7}$$

One way to formulate this optimization problem is to convert this PI to one that weights the states and inputs but has a state/input cross-weighting term [simply substitute (5.4-29) into J]. This optimization problem is solved by Lewis (1986).

An alternative (see the problems) is to minimize

$$J = \tfrac{1}{2}\mathrm{tr}\left[P\dot{\tilde{x}}(0)\dot{\tilde{x}}^{\mathrm{T}}(0)\right], \tag{5.5-8}$$

with P the solution to

$$A_c^{\mathrm{T}}P + PA_c + Q = 0 \tag{5.5-9}$$

Again, any optimization technique may be used. More details on this formulation may be found in the work of Quintana et al. (1976).

In the step-response shaping problem, the value of the initial state derivative vector to use in (5.5-8) is easy to determine since $x(0) = 0$ and $\tilde{x}$ is a constant, so that according to (5.4-16) and (5.4-15)

$$\dot{\tilde{x}}(0) = B_c r_0 \tag{5.5-10}$$

Time-Dependent Weighting

One final form of the PI remains to be discussed. A step response that is apparently good (i.e., fast, with acceptable overshoot and settling time) may contain a slow pole(s) with small residue, so that the response creeps for a long time as it nears its final value. The quadratic performance criterion penalizes small errors relatively lightly and so does not tend to suppress this kind of behavior.

Thus, in the spirit of the classical (ITAE, ISTSE, etc.) performance indices (D'Azzo and Houpis, 1988) we define a PI that contains a time-weighted component:

$$J = \tfrac{1}{2}\int_0^\infty \left(t^k\tilde{x}^{\mathrm{T}}P\tilde{x} + \tilde{x}^{\mathrm{T}}Q\tilde{x}\right) dt \tag{5.5-11}$$

If we are interested in including a control-weighting term $\tilde{u}^{\mathrm{T}}R\tilde{u}$ in (5.5-11) and in using the output feedback (5.5-2), we may add the term $C^{\mathrm{T}}K^{\mathrm{T}}RKC$ (since $\tilde{u}^{\mathrm{T}}R\tilde{u} = \tilde{x}^{\mathrm{T}}C^{\mathrm{T}}K^{\mathrm{T}}RKC\tilde{x}$) to the appropriate state-weighting matrix P or Q, depending on whether we wish to multiply the control-weighting term by t^k. For instance, if the control input term is not to be weighted by t^k, the PI (5.5-11) takes on the form

$$J = \tfrac{1}{2}\int_0^\infty \left[t^k\tilde{x}^{\mathrm{T}}P\tilde{x} + \tilde{x}^{\mathrm{T}}\left(Q + C^{\mathrm{T}}K^{\mathrm{T}}RKC\right)\tilde{x}\right] dt \tag{5.5-12}$$

If it is desired to have the control weighting multiplied by t^k, the term $C^{\mathrm{T}}K^{\mathrm{T}}RKC$ should be added to P instead of Q.

Whether or not the control effort should be time weighted is a matter for experiment with the particular design. The time-varying weighting in the PI places a heavy penalty on errors that occur late in the response and is thus very effective in suppressing the effect of a slow pole as well as in eliminating lightly damped settling behavior.

Due to the factor t^k, the optimal gain K that minimizes J is time varying. However, to obtain useful designs we will determine the suboptimal solution that assumes a time-invariant control gain K. Note that time-varying gains would be very difficult to gain schedule.

We may successively integrate by parts (see the problems) to show that the value of (5.5-12) for a given value of K is given by successively solving the nested Lyapunov equations

$$\begin{aligned} 0 &= g_0 \equiv A_c^{\mathrm{T}}P_0 + P_0A_c + P \\ 0 &= g_1 \equiv A_c^{\mathrm{T}}P_1 + P_1A_c + P_0 \\ &\vdots \\ 0 &= g_{k-1} \equiv A_c^{\mathrm{T}}P_{k-1} + P_{k-1}A_c + P_{k-2} \\ 0 &= g_k \equiv A_c^{\mathrm{T}}P_k\mathrm{A}_c + k!P_{k-1} + Q + C^{\mathrm{T}}K^{\mathrm{T}}RKC \end{aligned} \tag{5.5-13}$$

Then

$$J = \tfrac{1}{2}\tilde{x}^{\mathrm{T}}(0)P_k\tilde{x}(0) = \tfrac{1}{2}\bar{x}^{\mathrm{T}}P_k\bar{x} = \tfrac{1}{2}\mathrm{tr}(P_kX) \tag{5.5-14}$$

A minimization routine such as Simplex (Nelder and Mead, 1964; Press et al., 1986) can be used to find the optimal gains using (5.5-13) and (5.5-14) to evaluate the PI for a specified value of the gain K.

Alternatively, to use a faster gradient-based routine, we may determine the gradient of J with respect to K. To do so, define the Hamiltonian

$$\mathscr{H} = \tfrac{1}{2}\mathrm{tr}(P_kX) + \tfrac{1}{2}\mathrm{tr}(g_0S_0) + \ldots + \tfrac{1}{2}\mathrm{tr}(g_kS_k), \tag{5.5-15}$$

where $\mathrm{S_i} \geq 0$ are matrices of undetermined Lagrange multipliers. Then, by differentiating $\mathscr{H}$ with respect to all variables, necessary conditions for a minimum may be found (see the Problems). These design equations for the LQ tracker with time weighting are summarized in Table 5.5-1.

To use a gradient-based optimization routine such as Davidon-Fletcher-Powell (Press et al., 1986), we may proceed as follows. For a given K, solve the nested Lyapunov equations for P_i and S_i. Since the g_i are then all zero, (5.5-15) shows that $J = \mathscr{H}$. Then (5.5-23) gives the gradient of J with respect to K, which is used by the gradient-based routine to find the updated value of K.

If it is desired to use LQ *regulator* design (as opposed to tracker design, that is, Table 5.3-1) with time-dependent weighting, one need only set $X = I$ (assuming that $E\{x(0)x^{\mathrm{T}}(0)\} = I$) and $\bar{x} = 0$ in the tracker design equations of Table 5.5-1.

TABLE 5.5-1 LQ Tracker with Time-Weighted PI

System Model

$$\dot{x} = Ax + Bu + Gr$$
$$y = Cx + Fr$$

Control

$$u = -Ky$$

Performance Index

$$J = \tfrac{1}{2}\int_0^\infty \left[t^k \tilde{x}^T P \tilde{x} + \tilde{x}^T \left(Q + C^T K^T RKC\right)\tilde{x}\right] dt$$

Optimal Output Feedback Control

$$\begin{aligned}
0 &= g_0 \equiv A_c^T P_0 + P_0 A_c + P \\
0 &= g_1 \equiv A_c^T P_1 + P_1 A_c + P_0 \\
&\ \vdots \\
0 &= g_{k-1} \equiv A_c^T P_{k-1} + P_{k-1} A_c + P_{k-2} \\
0 &= g_k \equiv A_c^T P_k + P_k A_c + k! P_{k-1} + Q + C^T K^T RKC
\end{aligned} \tag{5.5-21}$$

$$\begin{aligned}
0 &= A_c S_k + S_k A_c^T + X \\
0 &= A_c S_{k-1} + S_{k-1} A_c^T + k! S_k \\
0 &= A_c S_{k-2} + S_{k-2} A_c^T + S_{k-1} \\
&\ \vdots \\
0 &= A_c S_0 + S_0 A_c^T + S_1
\end{aligned} \tag{5.5-22}$$

$$0 = \frac{\partial \mathscr{H}}{\partial K} = RKCS_k C^T - B^T(P_0 S_0 + \cdots + P_k S_k)C^T + B^T A_c^{-T} P_k \overline{x}\,\overline{y}^T \tag{5.5-23}$$

with r a unit step of magnitude r_0 and

$$\overline{x} = -A_c^{-1} B_c r_0 \tag{5.5-24}$$

$$\overline{y} = C\overline{x} + Fr_0 \tag{5.5-25}$$

$$X = \overline{x x}^T = A_c^{-1} B_c r_0 r_0^T B_c^T A_c^{-T} \tag{5.5-26}$$

where

$$A_c = A - BKC, \quad B_c = G - BKF$$

Optimal Cost

$$J = \tfrac{1}{2}\mathrm{tr}(P_k X)$$

Software to determine the optimal value of K given the design parameters k, Q, and R (for both the regulator and tracker) is described in Appendix B. It is called program LQ.

A combination of derivative and time-dependent weighting occurs in the PI:

$$J = \frac{1}{2}\int_0^\infty \left(t^k \dot{\tilde{x}}^{\mathrm{T}} P \dot{\tilde{x}} + \dot{\tilde{x}}^{\mathrm{T}} \;\; Q\dot{\tilde{x}}\right) dt \tag{5.5-16}$$

The optimal gains in this situation may be determined by minimizing

$$J = \frac{1}{2}\dot{\tilde{x}}^{\mathrm{T}}(0) P_k \dot{\tilde{x}}(0) = \frac{1}{2} r_0^{\mathrm{T}} B_c^{\mathrm{T}} P_k B_c r_0 \tag{5.5-17}$$

subject to (5.5-13) with $R = 0$.

A Fundamental Design Property

We now mention a fact of key importance in connection with time-dependent weighting. We will be very concerned to use PIs that are sensible from a design point of view. That is, we will not be content to select P and Q in Table 5.5-1 as $n \times n$ matrices and juggle their entries until a suitable design occurs. This sort of approach is one of the fundamental flaws of modern LQ design.

A sensible PI is one of the form

$$J = \frac{1}{2}\int_0^\infty \left(t^k \tilde{e}^{\mathrm{T}} \tilde{e} + r\tilde{u}^{\mathrm{T}} \tilde{u}\right) dt, \tag{5.5-18}$$

where, according to Section 5.4, the error deviation is given by

$$\tilde{e} = -H\tilde{x}, \tag{5.5-19}$$

with $z = Hx$ the performance output. This PI corresponds to our desire to make the error small without too much control energy. Since $\tilde{e}^{\mathrm{T}}\tilde{e} = \tilde{x}^{\mathrm{T}} H^{\mathrm{T}} H\tilde{x}$, it amounts to using the PI in Table 5.5-1 with $Q = 0$, $R = rI$, and $P = H^{\mathrm{T}}H$.

However, if (H, A) is not observable and if $k = 0$, there may be problems with any LQ design (Lewis, 1986). Specifically, in this case the Lyapunov equation

$$A_c^{\mathrm{T}} P + PA_c + H^{\mathrm{T}} H + C^{\mathrm{T}} K^{\mathrm{T}} RKC = 0 \tag{5.5-20}$$

may not have a positive-definite solution P. This could result in some of the feedback gains being set to zero in the LQ optimal solution.

To correct this, we could add a term like $\tilde{x}^{\mathrm{T}} Q\tilde{x}$ in the PI, with $(\sqrt{Q}, A)$ observable. This, however, is exactly what we are trying to avoid, since it will give us all of the elements of Q as design parameters that should be varied until a suitable K

results. To avoid this counterintuitive approach, we need only select $k > 0$ in the PI in Table 5.5-1. To see why, consider the case $Q = 0$ and examine Table 5.5-1. Note that even if $(\sqrt{P}, A)$ is not observable, $[(k!P_{k-1})^{1/2}, A]$ may be observable for some $k > 0$. If so, the last Lyapunov equation in (5.5-21) will have a positive-definite solution P_k, which will correct the observability problem. That is, by using time weighting, the LQ observability problem is corrected. We will illustrate this point in Example 5.5-2.

Example 5.5-1: Constrained Feedback Control for F-16 Lateral Dynamics In Example 5.3-1 we showed how to design a lateral stability augmentation system for an F-16. The resulting gain matrix K had eight nonzero entries. It would be desirable to avoid gain scheduling such a large number of gains as well as to avoid feedback from roll rate and bank angle to rudder and from washed-out yaw rate and sideslip to aileron. That is, the gain matrix should have the form

$$K = \begin{bmatrix} 0 & x & 0 & x \\ x & 0 & x & 0 \end{bmatrix} \tag{1}$$

This *constrained output feedback* regulator is quite easy to design using the techniques just discussed. Indeed, select a PI of the form (5.5-5) with $g_{11} = 1000$, $g_{13} = 1000$, $g_{12} = 1000$, $g_{24} = 1000$ in order to weight the unwanted entries of $K = [k_{ij}]$. Then the algorithm of Table 5.3-1, with the modified equation (5.5-6) used to evaluate the PI in a numerical minimization scheme, yields the feedback gain matrix

$$K = \begin{bmatrix} -1\text{E}-3 & -0.55 & 1\text{E}-3 & -0.49 \\ -1.14 & -1\text{E}-3 & 0.05 & 1\text{E}-3 \end{bmatrix} \approx \begin{bmatrix} 0 & -0.55 & 0 & -0.49 \\ -1.14 & 0 & 0.55 & 0 \end{bmatrix} \tag{2}$$

The same Q and R were used as in Example 5.3-1. The resulting closed-loop poles are

$$\begin{aligned} s = -1.16 \pm j0.99 \qquad & \text{dutch roll mode}\,(r, \beta) \\ -0.79 \qquad & \text{spiral mode} \\ -7.42 \qquad & \text{roll subsidence mode} \\ -11.54 \pm j19.51, -12.27 \qquad & \end{aligned} \tag{3}$$

■

Note that the spiral and roll subsidence modes now consist of two real poles so that the complex roll mode is absent. The closed-loop response is shown in Figure 5.5-1. It should be compared to the response obtained in Example 5.3-1 as well as in examples in Chapter 4.

An alternative design technique is simply to use the option in program LQ of instructing the program to leave certain elements of K fixed at zero during the minimization procedure.

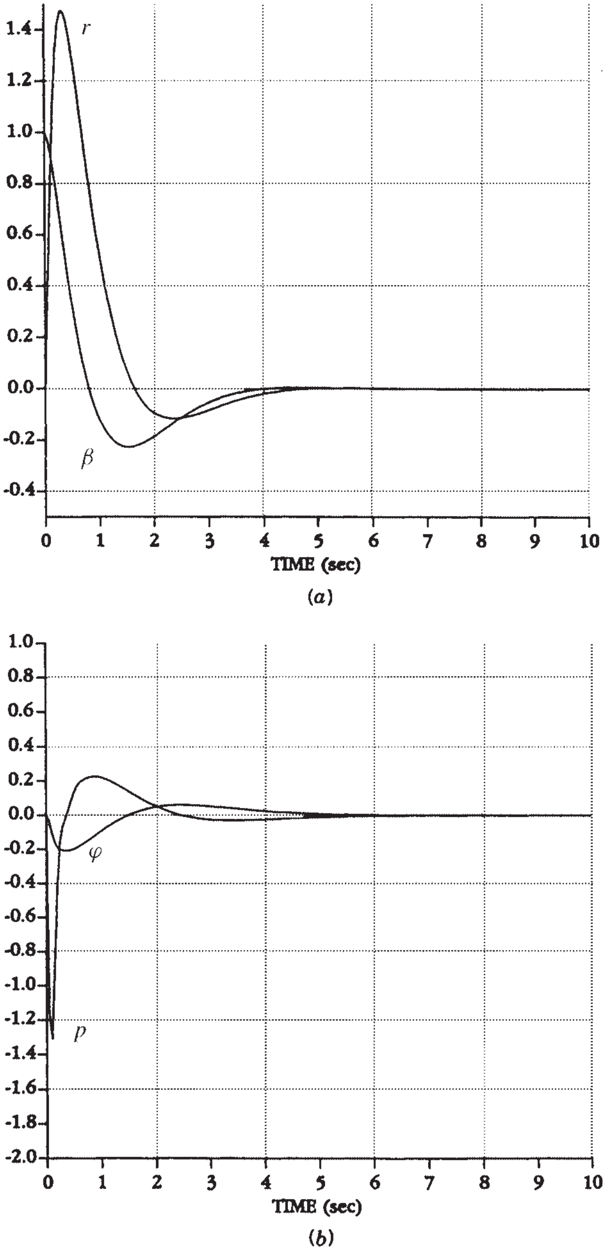


Figure 5.5-1 Closed-loop lateral response: (*a*) dutch roll states β and r; (*b*) spiral and roll subsidence states ϕ and p.

Example 5.5-2: Time-Dependent Weighting Design of Normal Acceleration CAS In Example 5.4-1 we designed a normal acceleration CAS. A deficiency with that approach was the need to check for the observability of $(\sqrt{Q}, A)$; there, unobservability led us to weight the integrator output in Q. In this example we show how to avoid the observability issue by using time-dependent weighting in the PI.

The aircraft and controller dynamics are the same as in Example 5.4-1. Here, however, we will select the time-weighted PI

$$J = \tfrac{1}{2}\int_0^\infty (t^2\tilde{e}^2 + \rho\tilde{u}^2)dt, \tag{1}$$

which is entirely sensible from a performance point of view and contains only one design parameter to be tuned. This corresponds to the PI in Table 5.5-1 with $P = H^T H$, $Q = 0$, $R = \rho$.

Selecting $\rho = 0.05$ and using program LQ, we obtained the control gains

$$K = [-0.847 \quad -0.452 \quad 1.647 \quad 8.602], \tag{2}$$

the closed-loop poles

$$\begin{aligned} s = &-1.90 \pm j2.58 \\ &-2.35 \\ &-13.88 \pm j13.12, \end{aligned} \tag{3}$$

and the step response shown in Figure 5.5-2. It is much better than the result of Example 5.4-1 and was obtained without juggling the elements of the Q-matrix or

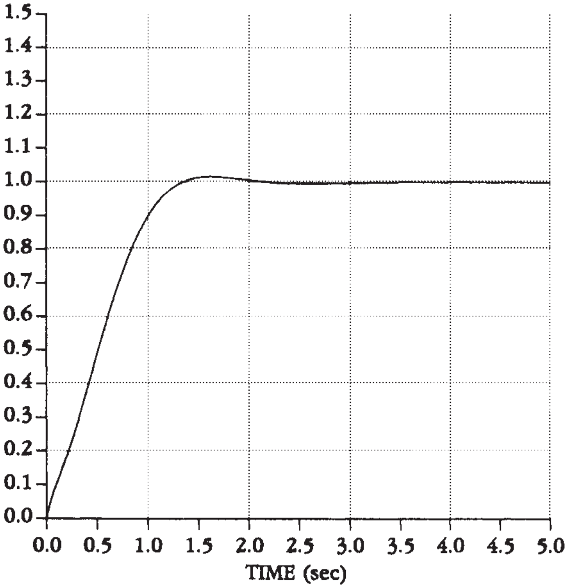


Figure 5.5-2 Normal acceleration step response.

worrying about observability issues. By using time weighting in the PI, we have formulated a design problem that has only one design parameter that needs to be varied, namely, the control weighting, ρ. This entire design took 5 minutes. Contrast to Example 4.5-3. ■

Example 5.5-3: Pitch-Rate Control System Using LQ Design In this example we reconsider pitch-rate control system design using LQ techniques. The approach to be used here should be compared to the classical approach used in Chapter 4. It will be demonstrated how two of the PIs just developed can simplify the control system design, since they have *only one design parameter that must be tuned to obtain good performance*. This LQ technique is therefore in sharp contrast to the classical approach, where we had to vary all three elements of the gain matrix in successive-loop-closure design. It is also in contrast to the traditional modern LQ approaches, where all the elements of the PI weighting matrices must generally be tuned to obtain good performance and where the observability properties of the PI must be considered in selecting the state-weighting matrix.

Since we are using a modern LQ-based approach, a sensible formulation of the problem should result in closed-loop stability for all selections of the design parameter. This is an extremely important property of modern control design techniques and in complete contrast to classical techniques, where stability in multiloop systems can be difficult to achieve.

(a) *Aircraft and Control System Dynamics*. The pitch control system is shown in Figure 5.5-3, where the control input is elevator actuator voltage $u(t)$ and r is a reference step input corresponding to the desired pitch command. Thus, the performance output, $z(t)$, is the pitch rate, q. The measured outputs $y(t)$ are pitch, q, and angle of attack, α; however, since α measurements are quite noisy, a low-pass filter with a

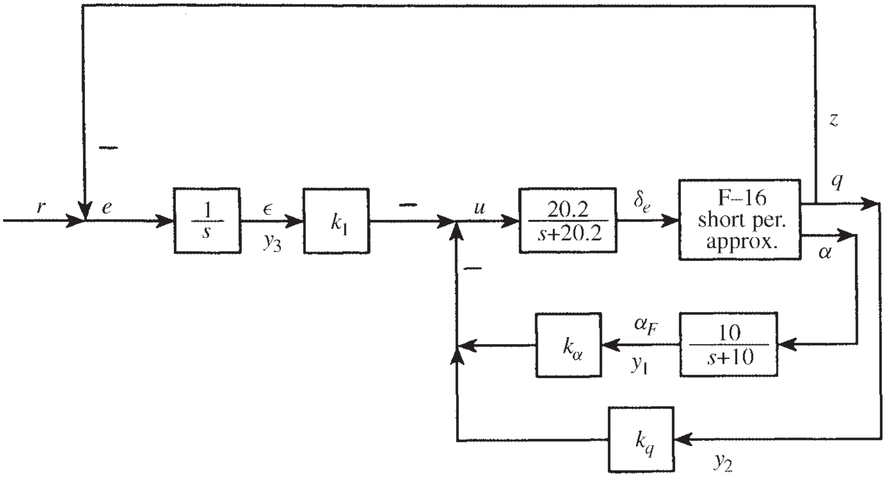


Figure 5.5-3 Pitch-rate control system.

cutoff frequency of 10 rad/s is used to provide filtered measurements α_F of the angle of attack. To ensure zero steady-state error an integrator was added in the feedforward channel; this corresponds to the compensator dynamics. The integrator output is ϵ.

We used the short-period approximation to the F-16 dynamics linearized about the nominal flight condition in Table 3.6-3 (502 ft/s, 0 ft altitude, level flight, with the cg at $0.35\overline{c}$). Thus, the basic aircraft states of interest are α and q. An additional state is introduced by the elevator actuator. The elevator deflection is δ_e.

The states and outputs of the plant plus compensator are

$$x = \begin{bmatrix} \alpha \\ q \\ \delta_e \\ \alpha_F \\ \epsilon \end{bmatrix}, \qquad y = \begin{bmatrix} \alpha_F \\ q \\ \epsilon \end{bmatrix} \tag{1}$$

and the system dynamics are described by

$$\dot{x} = Ax + Bu + Gr \tag{2}$$

$$y = Cx + Fr \tag{3}$$

$$z = Hx \tag{4}$$

with

$$A = \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 & 0 & 0 \\ 0.82225 & -1.07741 & -0.17555 & 0 & 0 \\ 0 & 0 & -20.2 & 0 & 0 \\ 10 & 0 & 0 & -10 & 0 \\ 0 & -57.2958 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 20.2 \\ 0 \\ 0 \end{bmatrix} \qquad G = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 57.2958 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad F = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$H = \begin{bmatrix} 0 & 57.2958 & 0 & 0 & 0 \end{bmatrix}$$

The factor of 57.2958 is added to convert angles from radians to degrees.

The control input is

$$u = -Ky = -[k_\alpha \quad k_q \quad k_I]y = -k_\alpha \alpha_F - k_q q - k_I \epsilon \tag{5}$$

It is required to select the feedback gains to yield good closed-loop response to a step input at r, which corresponds to a single-input/multi-output design problem.

Now consider two LQ designs based on two different performance indices. The modified PIs introduced in this section will mean that we do not need to worry about observability issues and that *only one design parameter will appear*. This is significant in view of the fact that there are five states and three control gains to find.

Since the integrator makes the system type 1, the steady-state error $\overline{e}$ is equal to zero and

$$e(t) = \widetilde{e}(t) \tag{6}$$

Thus, the PI term involving $\overline{e}$ in Section 5.4 is not required.

(b) *Time-Dependent Weighting Design*. Consider the PI

$$J = \frac{1}{2}\int_0^\infty (t^2\widetilde{e}^2 + \rho\widetilde{u}^2)dt \tag{7}$$

This is a natural PI that corresponds to the actual performance requirements of keeping the tracking error small without using too much control energy and also has the important advantage of requiring the adjustment of only one design parameter ρ. It amounts to using $P = H^T H$, $Q = 0$, $R = \rho$ in Table 5.5-1.

Program LQ was used to solve the design equations in Table 5.5-1 for several values of ρ. A good step response was found with $\rho = 1$, which yielded optimal gains of

$$K = [-0.046 \quad -1.072 \quad 3.381] \tag{8}$$

closed-loop poles of $s = -8.67 \pm j9.72$, -9.85, -4.07, and -1.04, and the step response in Figure 5.5-4. Compare to the results of Example 4.5-1.

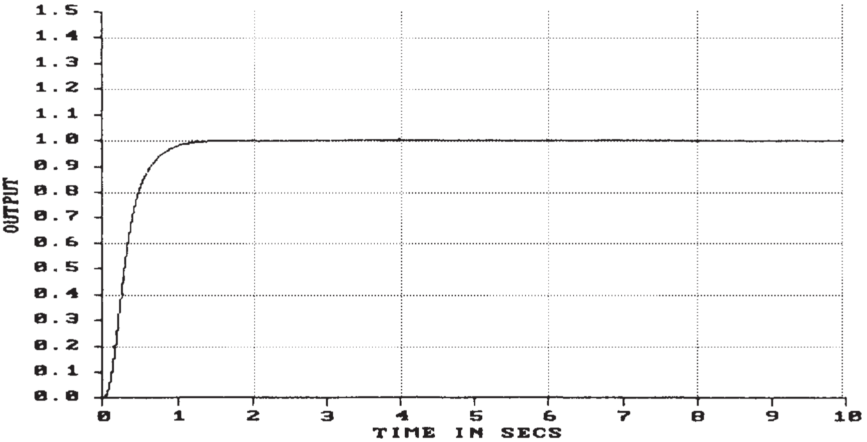


Figure 5.5-4 Pitch-rate step response using time-dependent weighting design.

(c) *Derivative Weighting Design.* Since elevator actuator *rate* has a stronger intuitive connection to "control activity" than does elevator displacement, let us illustrate derivative weighting by repeating the design. Select the PI

$$J = \tfrac{1}{2}\int_0^\infty (pt^2e^2 + \dot{\delta}_e^2)dt \tag{9}$$

Since e$(t) = \dot{\epsilon}(t)$, this may be written

$$J = \tfrac{1}{2}\int_0^\infty (pt^2\dot{\epsilon}^2 + \dot{\delta}_e^2)dt, \tag{10}$$

with $\epsilon(t)$ and $\delta_e(t)$ the deviations in the integrator output and elevator deflection. This is exactly the derivative weighting PI (5.5-16) with $P = \text{diag}\{0, 0, 0, 0, p\}$ and $Q = \text{diag}\{0, 0, 1, 0, 0\}$.

It should be emphasized that we have again been careful to formulate the problem in such a way that only one design parameter, namely, p, needs to be adjusted in the iterative design phase.

The software described in Appendix B was used to minimize (5.5-17) subject to (5.5-13) for several values of p. The weight $p = 10$ led to a good step response, as shown in Figure 5.5-5. The feedback gain matrix was

$$K = [-0.0807 \quad -0.475 \quad 1.361] \tag{11}$$

and the closed-loop poles were at $s = -3.26 \pm j2.83$, -1.02, -10.67, and -14.09. These poles are virtually identical to those obtained in Example 4.5-1. Compare the design process in this example with the design process in that example.

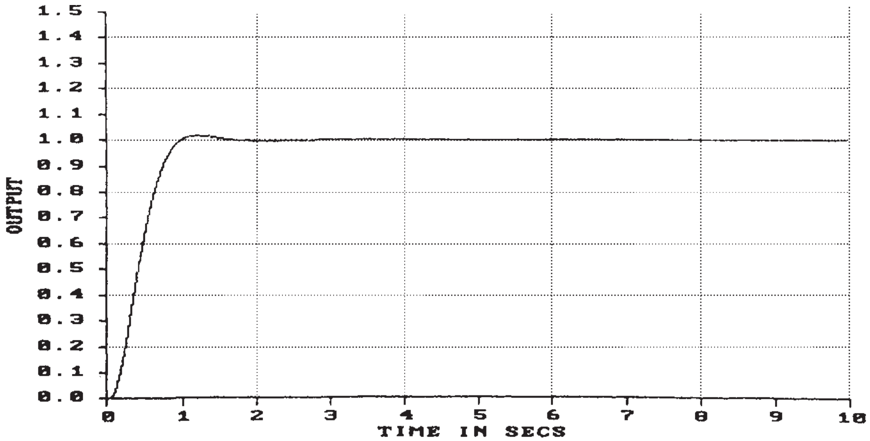


Figure 5.5-5 Pitch-rate step response using derivative weighting design.

■

Example 5.5-4: Multivariable Wing Leveler In this example, we will illustrate a multi-input/multi-output (MIMO) design using the LQ approach developed in this chapter. This example should be compared with Chapter 4, where we designed a two-input/two-output roll damper/yaw damper using classical control by successive loop closures.

(a) *Control System Structure.* The control system shown in Figure 5.5-6 is meant to hold the aircraft's wing level while providing yaw damping by holding washed-out yaw rate, r_w, at zero. It is a two-channel system. In the upper channel there is an outer-loop unity-gain feedback of bank angle, ϕ, with an inner-loop feedback of roll rate, p. This channel has a PI compensator to make the system type 1 to achieve zero steady-state bank angle error. The control input for the upper channel is aileron deflection, δ_a. The lower channel has a feedback of washed-out yaw rate, r_w; in this channel the control input is rudder deflection, δ_r.

The reference command is $r_c = [r_\phi \quad r_r]^{\mathrm{T}}$. The tracking control system should hold ϕ at the commanded value of r_ϕ, and r_w at the commanded value of r_r, which is equal to zero. To hold the wing level, r_ϕ is set equal to zero, although it could be any commanded bank angle. The tracking error is $e = [e_\phi \quad e_r]^{\mathrm{T}}$ with

$$\begin{aligned} e_\phi &= r_\phi - \phi \\ e_r &= r_r - r_w \end{aligned} \tag{1}$$

(b) *State Equations for Aircraft and Control Dynamics.* As in Example 5.3-1, we used the F-16 linearized lateral dynamics at the nominal flight condition in Table 3.6-3 ($V_T = 502\,\mathrm{ft/s}$, 300 psf dynamic pressure, cg at $0.35\bar{c}$) retaining the lateral states sideslip, β, bank angle, ϕ, roll rate, p, and yaw rate, r. Additional states δ_a and δ_r are introduced by the aileron and rudder actuators. The washout filter state is called x_w.

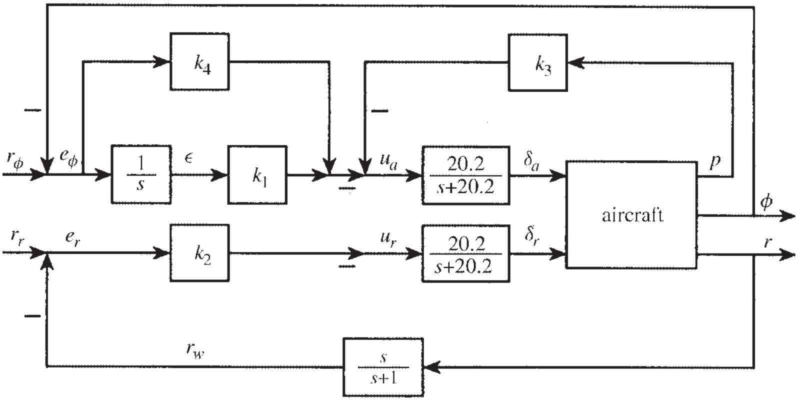


Figure 5.5-6 Wing-leveler lateral control system.

We denote by ϵ the output of the controller integrator in the upper channel. Thus, the entire state vector is

$$x = [\beta \quad \phi \quad p \quad r \quad \delta_a \quad \delta_r \quad x_w \quad \epsilon]^{\mathrm{T}} \tag{2}$$

The full state-variable model of the aircraft plus actuators, washout filter, and control dynamics is of the form

$$\dot{x} = Ax + Bu + Gr_c \tag{3}$$

with

$$A = \begin{bmatrix} -0.3220 & 0.0640 & 0.0364 & -0.9917 & 0.0003 & 0.0008 & 0 & 0 \\ 0 & 0 & 1 & 0.0037 & 0 & 0 & 0 & 0 \\ -30.6492 & 0 & -3.6784 & 0.6646 & -0.7333 & 0.1315 & 0 & 0 \\ 8.5395 & 0 & -0.0254 & -0.4764 & -0.0319 & -0.0620 & 0 & 0 \\ 0 & 0 & 0 & 0 & -20.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -20.2 & 0 & 0 \\ 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20.2 & 0 \\ 0 & 20.2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \tag{4}$$

The performance output that should follow the reference input $[r_\phi \quad r_r]^{\mathrm{T}}$ is

$$z = \begin{bmatrix} \phi \\ r_w \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 & 0 \end{bmatrix} x = Hx, \tag{5}$$

where the factor 57.2958 converts radians to degrees. According to the figure, if we define the measured output as

$$y = \begin{bmatrix} \epsilon \\ e_r \\ p \\ e_\phi \end{bmatrix} = Cx + Fr_c \tag{6}$$

with

$$c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -57.2958 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \tag{7}$$

the control input $u = [u_a \quad u_r]^{\mathrm{T}}$ may be expressed as

$$u = -Ky \tag{8}$$

with

$$K = \begin{bmatrix} k_1 & 0 & k_3 & k_4 \\ 0 & k_2 & 0 & 0 \end{bmatrix} \tag{9}$$

The control gains, k_i, must now be determined for satisfactory closed-loop response. Therefore, this is an output feedback design problem exactly of the form addressed in this chapter. Note that some of the entries of K must be constrained to zero to yield the desired control structure shown in Figure 5.5-6.

(c) *LQ Output Feedback Design.* To guarantee tracking by $z(t)$ of the reference command $r_c(t)$, we may select the PI

$$J = \tfrac{1}{2} \int \left(t^2 \tilde{x}^{\mathrm{T}} P \tilde{x} + \tilde{u}^{\mathrm{T}} \tilde{u} \right) dt + \tfrac{1}{2} v \bar{e}^{\mathrm{T}} \bar{e} \tag{10}$$

with $\tilde{x}(t)$ and $\tilde{u}(t)$ the state and control deviations defined in Section 5.4 and $\bar{e}$ the steady-state error. Although the integrator in the upper control channel guarantees that $\bar{e}_\phi$ will be zero, the steady-state error weighting v is required to ensure that $\bar{e}_r$ is small. Note that v is a scalar.

The design equations for K using this PI are given in Table 5.5-1, with, however, the extra terms from Table 5.4-1 added to (5.5-23) due to the steady-state error weighting v. Thus, K is easily determined using program LQ.

Several attempts were made to obtain suitable closed-loop behavior using different values for v and P. Finally, it was found that good behavior was obtained with $v = 10$ and P selected to weight the states β, ϕ, p, r and ϵ, as well as the cross-term in ϕr. That is,

$$p_{11} = p_{22} = p_{33} = p_{44} = p_{88} = 100, \quad p_{24} = p_{42} = 10 \tag{11}$$

The motivation for the p_{24} cross-weighting is that, after a few design attempts with different P, it was found that there were always several barely stable and badly damped complex pole pairs in the closed-loop system. The p_{24}, p_{42} cross-weighting penalizes the dutch roll mode, which was one of the ones yielding problems. The motivation for selecting p_{88} weighting is that good results are generally obtained if the integrator output is weighted.

Using the final selection of v and P, the control gains were found to be

$$\begin{aligned} k_1 &= 15.04, & k_2 &= 0.1822, \\ k_3 &= -5.348, & k_4 &= 22.52, \end{aligned} \tag{12}$$

yielding closed-loop poles of

$$\begin{gathered} -0.72 \pm j3.03 \\ -1.12 \pm j0.07 \\ -2.43, -5.05 \\ -15.3, -19.4 \end{gathered} \tag{13}$$

(d) *Simulation*. The closed-loop response to a reference command of $r_\phi = 1$, $r_r = 0$ is shown in Figure 5.5-7. The transient response and steady-state errors are both quite satisfactory. This is despite the presence of an underdamped pole pair at $-0.72 \pm j3.03$. One should recall the discussion in Chapter 4, where the strong

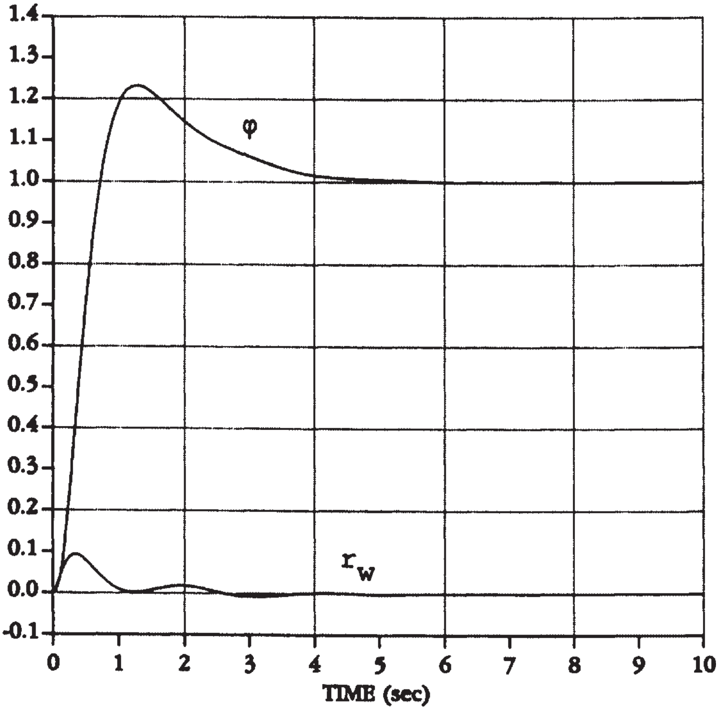


Figure 5.5-7 Closed-loop response to a command of $r_\phi = 1$, $r_r = 0$. Bank angle ϕ (rad) and washed-out yaw rate (rad/s).

coupling between the aircraft roll and yaw channels was emphasized. Despite this, Figure 5.5-7 shows that we have been quite successful in decoupling the yaw rate from the bank angle. ■

Example 5.5-5: Glide-Slope Coupler A glide-slope coupler is part of an automatic landing system—it guides an aircraft down a predetermined flight path to the end of a runway. At the end of the descent another control system, the automatic flare control (Example 5.6-1), is switched in to cause the aircraft to flare to a landing.

In this example we design a glide-slope coupler for the longitudinal dynamics of a medium-sized transport aircraft. Our approach should be compared to the frequency-domain approach in Example 4.6-4. See also Blakelock (1965).

(a) *Aircraft Dynamics*. The important inputs are both elevator and throttle for this problem, since both are needed to fly down a glide path in a coordinated manner. Exactly as in Example 4.6-4, the longitudinal dynamics of the aircraft were linearized about a velocity of $V_T = 250\,\text{ft/s}$ with the cg at $0.25\bar{c}$ and including throttle and elevator actuators. The state and control inputs are

$$x = [v_t \quad \alpha \quad \theta \quad q \quad \delta_t \quad \delta_e]^{\mathrm{T}}, \quad u = [u_t \quad u_e]^{\mathrm{T}}, \tag{1}$$

with v_T the deviation from trim velocity. The dynamics are described by

$$\dot{x} = Ax + Bu, \tag{2}$$

where A and B may be found by referring to Example 4.6-4. (In finding the A and B in (2) from the matrices in Example 4.6-4, note our selection of states.)

At this point it is worthwhile to examine Figure 5.5-8, which we are starting to construct.

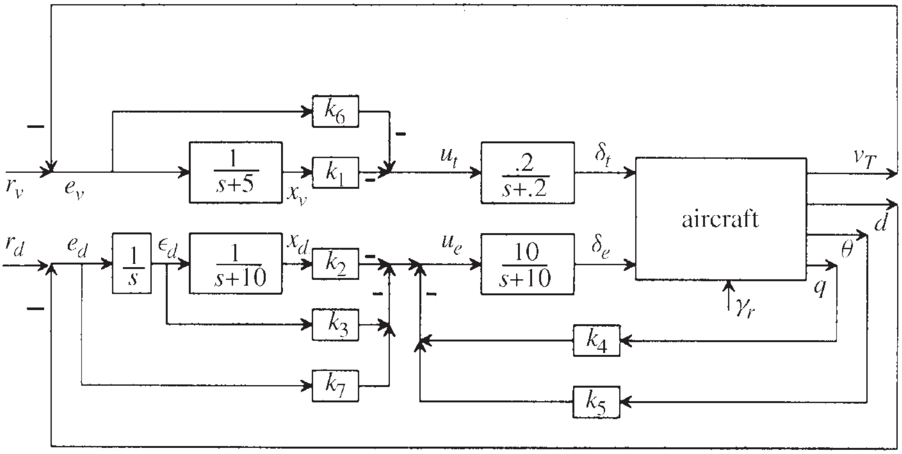


Figure 5.5-8 Glide-slope coupler.

(b) *Glide-Slope Geometry*. The glide-slope geometry is discussed in Example 4.6-4. The commanded or reference flight-path angle $-\gamma_r$ is generally 2.5°. The perpendicular distance from the glide path is $d(t)$.

Our control objectives in the glide-slope coupler are to regulate to zero the off-glide-path distance, d, and the deviation from trim velocity, v_t. Then the aircraft will remain on the glide path with the nominal velocity of $V_T = 250\,\text{ft/s}$. To accomplish this, the two control inputs are throttle and elevator. The outputs available for feedback are pitch rate, q, pitch angle, θ, v_T, and d, which is available from measurements taken from the ground.

The component of velocity perpendicular to the glide path is given by

$$\dot{d} = V_T \sin(\gamma - \gamma_r) \approx V_T(\gamma - \gamma_r) \tag{3}$$

when $(\gamma - \gamma_r)$ is small. We will assume that the velocity deviation v_T is small and take v_T in (3) as the trim velocity 250 ft/s. To follow the glide path, we require $d = 0$, so that the flight-path angle γ should be equal to γ_r. Then the aircraft will descend at an angle of $\gamma_r = -2.5°$.

In terms of variables in the state vector in (1), we may use $\gamma = \theta - \alpha$ to write

$$d = V_T\theta - V_T\alpha - \frac{V_T}{57.2958}\gamma_r = V_T\theta - V_T\alpha - 4.3633\gamma_r, \tag{4}$$

with θ and α in radians and γ_r in degrees. Therefore, we may include the off-glide-path distance d as a state in (1) by redefining

$$x = \begin{bmatrix} v_T & \alpha & \theta & q & d & \delta_t & \delta_e \end{bmatrix}^{\mathrm{T}} \tag{5}$$

(c) *Control System Structure*. Our objective is to regulate v_T and d to zero. Thus, we may define the performance output as

$$z = \begin{bmatrix} v_T \\ d \end{bmatrix} = Hx \tag{6}$$

Now examine Figure 5.5-8, which we have drawn to show that this may be considered as a tracking problem with reference commands r_v and r_d of zero. The tracking error is $e = [e_v \quad e_d]^{\mathrm{T}}$ with

$$\begin{aligned} e_v &= r_v - v_T \\ e_d &= r_d - d \end{aligned} \tag{7}$$

To obtain zero steady-state error in $v_T(t)$ and $d(t)$, we could add integrators in each of the forward error paths. However, according to the open-loop dynamics in Example 4.6-4 there are already several poles near the origin. Adding more poles near the origin makes the problem of stabilization more difficult.

Since we are more concerned about keeping d exactly zero, let us only add an integrator in the forward path corresponding to the tracking error in d. We can then obtain a small enough error in v_T without a forward-path integrator by using weighting of the steady-state error, as we will soon see.

An additional consideration for including a forward-path integrator in the d channel is the following. Note from (4) and Figure 5.5-8 that the commanded glide-path angle γ_r acts as a constant disturbance of magnitude $-2.5°$ into the system. The disturbance affects d. To reject this constant disturbance, we need a type-1 system with respect to d, which requires the integrator in the d feedforward path.

We can gain considerable insight by having root-locus design techniques in mind during a design by modern control. Thus, to pull the closed-loop poles into the left-half plane, we may add compensator zeros in the left-half plane. To implement the compensators without pure differentiation, we should add poles relatively far in the left-half plane, where they will not appreciably affect the root locus. Thus, let us propose a lead compensator in each forward channel (see Figure 5.5-8).

The compensators we propose are of the form

$$\begin{aligned} \frac{w_v}{e_v} &= \frac{k_1}{s+5} + k_6 = k_6 \frac{s + (5 + k_1/k_6)}{s+5} \\ u_t &= -w_v \end{aligned} \tag{8}$$

and

$$\begin{aligned} \frac{w_d}{e_d} &= \frac{k_2}{s(s+10)} \frac{k_3}{s} + k_7 \\ &= k_7 \frac{s^2 + (10 + k_3/k_7)s + (k_2 + 10k_3)/k_7}{s(s+10)} \\ u_e &= -w_d \end{aligned} \tag{9}$$

The important point to note is that, by varying the control gains, we may adjust *both the compensator gain and its zeros*. Thus, the *LQ optimization routine can adjust the zeros of the compensators*, presumably inducing lead compensation where it is required. We have selected the throttle compensator pole at $s = -5$ and the distance compensator pole at $s = -10$; however, any poles far to the left compared to the aircraft poles would suffice.

As we have seen in Example 4.6-4, selecting multiple control gains by classical techniques requires a successive-loop-closure approach. We hope to show that finding suitable gains using modern control theory is far easier, given a sensible problem formulation.

To formulate the controller so that the gains may be determined by our output feedback LQ approach, note that state-variable representations of (8) and (9) are given by

$$\dot{x}_v = -5x_v + e_v = -5x_v - v_T + r_v \tag{10}$$

$$u_t = -k_1 x_v - k_6 e_v = -k_1 x_v - k_6(v_T + r_v) \tag{11}$$

and

$$\dot{\epsilon}_d = e_d = -d + r_d \tag{12}$$

$$\dot{x}_d = -10x_d + \epsilon_d \tag{13}$$

$$u_e = -k_2 x_d - k_3 \epsilon_d - k_7 e_d = -k_2 x_d - k_3 \epsilon_d - k_7(-d + r_d) \tag{14}$$

The dynamical equations (4), (10), (12), and (13) may be incorporated into the system description by defining the augmented state

$$x = \begin{bmatrix} v_T & \alpha & \theta & q & d & \delta_t & \delta_e & x_v & x_d & \epsilon_d \end{bmatrix}^{\mathrm{T}} \tag{15}$$

Then the augmented system is described by

$$\dot{x} = Ax + Bu + Gr \tag{16}$$

with

$$A = \begin{bmatrix} -0.04 & 19.0096 & -32.1689 & 0 & 0 & 10.1 & 0 & 0 & 0 & 0 \\ -0.001 & -0.64627 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.7739 & 0 & -0.529765 & 0 & 0.02463 & -0.011 & 0 & 0 & 0 \\ 0 & -250 & 250 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -10 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & -5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -4.3633 \\ 0.2 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{17}$$

To incorporate the constant disturbance γ_r required in (4), we have defined an augmented input

$$u' = [u^{\mathrm{T}} \quad \gamma_r]^{\mathrm{T}} = [u_i \quad u_e \quad \gamma_r]^{\mathrm{T}} \tag{18}$$

Inputs such as γ_r, which are not actual controls, or reference signals $r(t)$ in the usual tracking system sense, are called *exogenous inputs*. Although they play the role of disturbances in the system, they are crucial in guaranteeing the desired system behavior.

Indeed, were we to ignore γ_r, the glide-slope coupler would always make the aircraft fly a horizontal path!

It should be clearly understood that for the design of the control system only the control input $u(t)$ is used. The full input $u'(t)$ will be required only in the simulation state, where γ_r will be set equal to $-2.5°$ to obtain the desired landing approach behavior.

In (16)/(17) the reference input is defined as

$$r = [r_v \quad r_d]^{\mathrm{T}}, \tag{19}$$

which is zero for the glide-slope coupler.

The equations (11) and (14) may be incorporated by defining a new measured output as

$$y = \begin{bmatrix} x_v & x_d & \epsilon_d & q & \theta & e_v & e_d \end{bmatrix}^{\mathrm{T}} \tag{20}$$

Then

$$y = Cx = Fr \tag{21}$$

with

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{22}$$

Now, according to Figure 5.5-8, the control vector $u(t)$ is given by the output feedback

$$u \begin{bmatrix} u_t \\ u_e \end{bmatrix} = - \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 & k_6 & 0 \\ 0 & k_2 & k_3 & k_4 & k_5 & 0 & k_7 \end{bmatrix} y = -Ky, \tag{23}$$

which has some elements constrained to zero.

According to (6), we may write

$$z = \begin{bmatrix} v_T \\ d \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x = Hx \tag{24}$$

At this point we have succeeded in casting the glide-slope coupler design problem into the formulation required in Tables 5.4-1 and 5.5-1.

It is important to understand the construction of the matrices in (17), (22), and (24), for this problem formulation stage is one of the most important phases in the LQ design technique.

(d) *PI and Control Design.* The other important phase in LQ design is the selection of an appropriate PI. Since the loop gain around the velocity loop is not of type 1, we will require weighting of the steady-state error to force v_T to go to zero at steady state. Thus, let us propose the PI

$$J = \tfrac{1}{2}\int_0^{\infty} (qt^2\tilde{e}^{\mathrm{T}}\tilde{e} + \tilde{u}^{\mathrm{T}}\tilde{u})dt + \tfrac{1}{2}v\bar{e}^{\mathrm{T}}\bar{e} \tag{25}$$

The motivation for the weighting t^2 follows. Weighting $\tilde{e}$ in the PI makes practical sense since we want it to vanish. However, $\tilde{e}^{\mathrm{T}}\tilde{e} = \tilde{x}^{\mathrm{T}}H^{\mathrm{T}}H\tilde{x}$, and (H, A) is not observable. In fact, the compensator states are not observable through $z = Hx$. An LQ design without the weighting t^2 would, therefore, fail. To correct the situation, we could weight the entire state in the PI by using a term like $\tilde{x}^T Q\tilde{x}$. However, this would give us too many design parameters (i.e., the elements of Q) and lead to a counterintuitive situation.

We prefer to work with sensible PIs, and in this situation we want to retain the weighting of $\tilde{e}(t)$, which is the variable of direct concern to us. Therefore, we use t^2 weighting to correct the observability problem. See the discussion preceding Example 5.5-1.

With t^2 weighting, a large value of the scalar q will result in a closed-loop system that is too fast. After several design iterations, it was found that suitable values for the PI design parameters were $q = 0.001$, $v = 100$. We employed program LQ to solve for the optimal gain K using the design equations of Table 5.5-1, including the steady-state error weighting from Table 5.4-1. We selected the option of fixing seven of the gain elements to zero as required by (23).

With $q = 0.001$, $v = 100$, the optimal control gains were

$$K = \begin{bmatrix} 2.598 & 0 & 0 & 0 & 0 & -0.9927 & 0 \\ 0 & 583.7 & -58.33 & -2.054 & -1.375 & 0 & 6.1 \end{bmatrix} \tag{26}$$

and the closed-loop poles were at

$$\begin{gathered} -0.27 \pm j1.01 \\ -0.36 \pm j0.49 \\ -0.37 \pm j.0.09 \\ -1.18, -4.78, -8.38, -10.09 \end{gathered} \tag{27}$$

Thus, the slowest time constant is $1/0.27 \approx 4\,\mathrm{s}$.

(e) *Simulation and Discussion.* A simulation of the glide-slope coupler appears in Figure 5.5-9. The aircraft was initialized in level flight at 1500 ft. The glide-slope coupler was switched on as the aircraft crossed through the glide path.

For simulation purposes, we used the exogenous input $\gamma_r = -2.5°$ (the desired glide-path angle) and reference commands of $r_v = 0$, $r_d = 0$. Altitude h was added as a state using the equation for vertical velocity

$$\dot{h} = V_T \sin\gamma \approx V_T(\theta - \alpha), \tag{28}$$

with V_T assumed to be the trim velocity of 250 ft/s.

According to the altitude plot in Figure 5.5-9*a*, after a small transient lasting about 20 s, the aircraft settles down onto the glide path and follows it down. Touchdown occurred at 137.5 s. Figure 5.5-9*b* shows the off-glide-path error d.

Figure 5.5-9*c* shows angle of attack and pitch angle. Note that after the transient the flight-path angle is given by $\gamma = \theta - \alpha = -2.5°$. Since in the descending configuration the aircraft is no longer at the original trim condition, a small angle of attack α of $-0.18°$ remains at steady state. The final pitch angle θ is $-2.68°$.

According to Figure 5.5-9*d*, the velocity deviation v_T settles out at 0.29 ft/s. This is a consequence of the fact that there is no integrator in the forward e_v path in Figure 5.5-8. Thus, the steady-state velocity on the glide path is $V_T = 250.29\,\text{ft/s}$; this is very suitable from a performance point of view. The smallness of the steady-state deviation despite the fact that the v_T loop is of type 0 is a consequence of the steady-state error weighting v in the PI (25).

Finally, the elevator and throttle control efforts δe and δt are shown in Figure 5.5-9*e*. Note the coordinated control achieved in this two-input system using the LQ approach. Since the descent down the glide path does not represent the original trim condition, the steady-state values of the control efforts are not zero. Intuitively, less throttle is required to maintain 250 ft/s if the aircraft is descending.

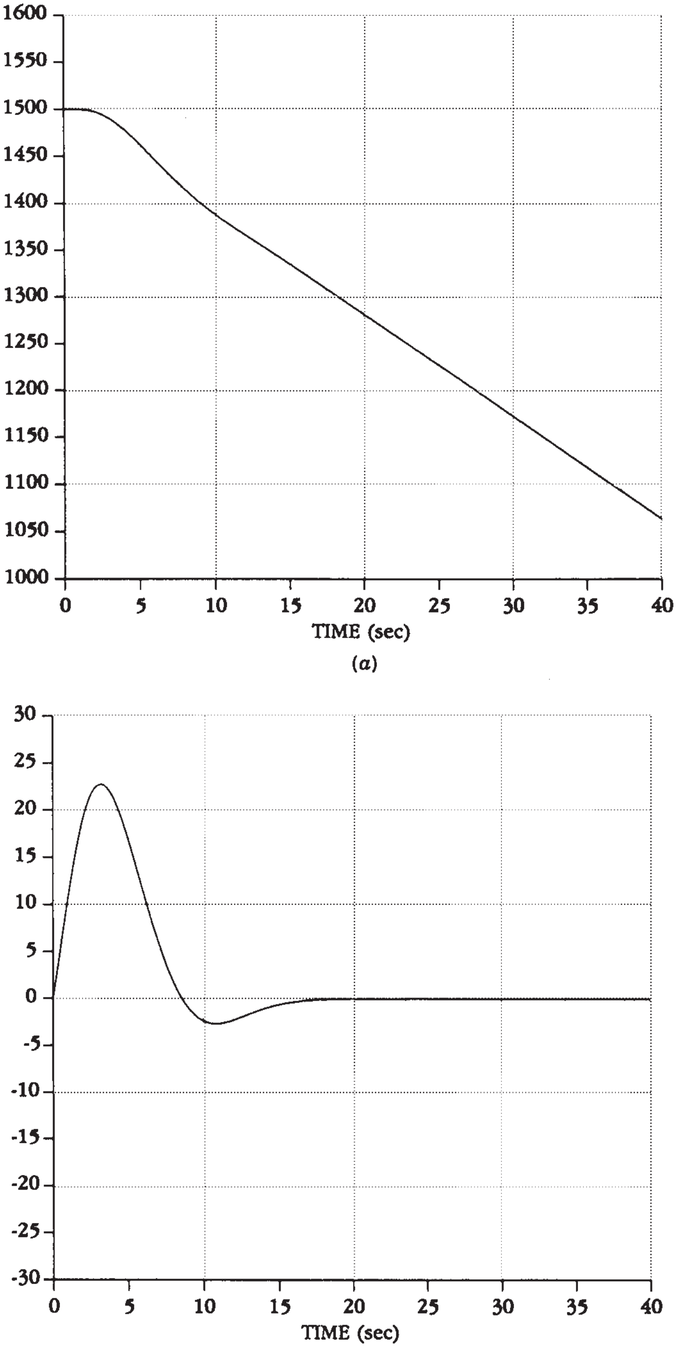
Figure 5.5-9 shows that, as the aircraft passes through the glide path, the elevator is pushed forward and the throttle is cut. As a result, the angle of attack and pitch angle decrease. After a slight positive position error, d, and an initial increase in velocity, v_T, further control effort stabilizes the aircraft on the glide path.

It is interesting to note the fundamental mechanism behind the glide-slope coupler. Namely, we regulate d in (3) and (4) to zero so that $\gamma = \gamma_r = -2.5°$. Then, according to (28), $\dot{h} = V_T \sin \gamma_r$, the appropriate descent rate to stay on the glide path.

With the optimal gains in (26), according to (8) the velocity channel compensator is

$$\frac{w_v}{e_v} = -0.9927\frac{s + 2.38}{s + 5}, \tag{29}$$

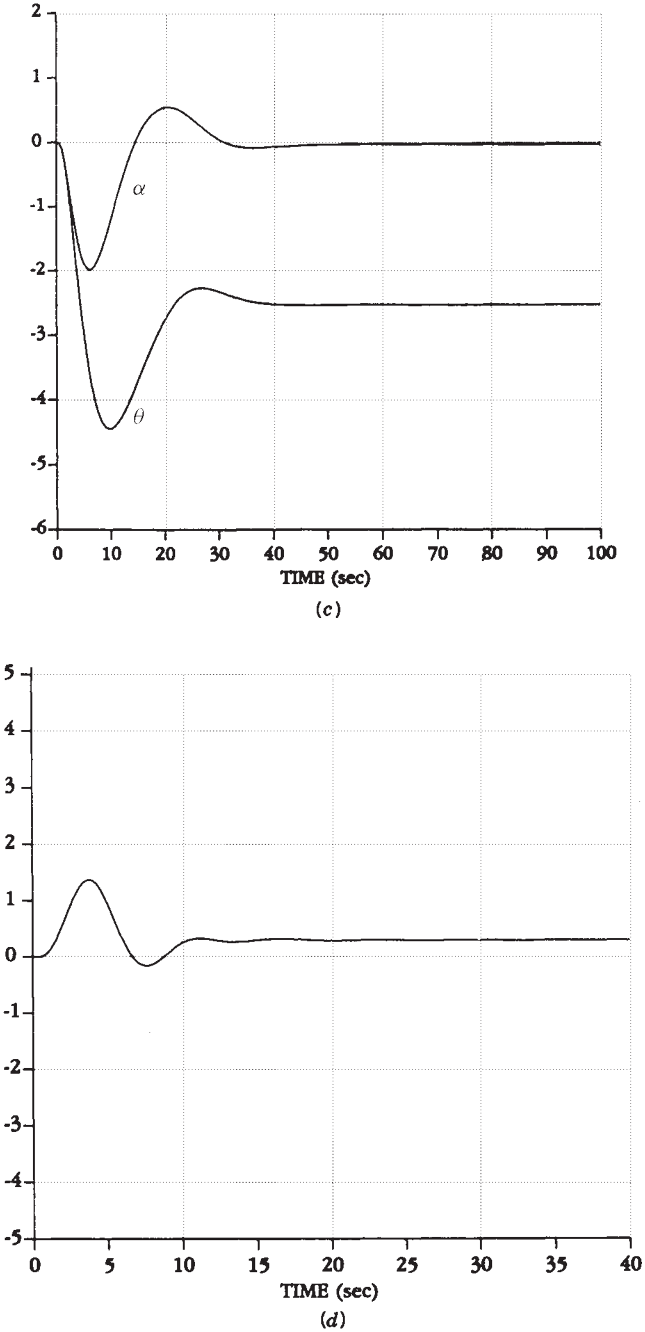
which is a lead compensator as anticipated. The zeros in the d channel compensator could similarly be found. It is important to note that our formulation has resulted in the *compensator zeros being selected in an optimal fashion*. This is an improvement over root-locus design, where the zeros are determined using the engineering judgment that actually only applies for single-input/single-output systems.



(*a*)

(*b*)

Figure 5.5-9 Glide-slope coupler responses: (*a*) altitude h (ft); (*b*) off-glide path distance d (ft). (*c*) Angle of attack α and pitch angle θ (deg); (*d*) velocity deviation $v_T(ft/s)$. (*e*) Control efforts δ_e (rad) and δ_i (per unit).



(*c*)

(*d*)

Figure 5.5-9 (*continued*)

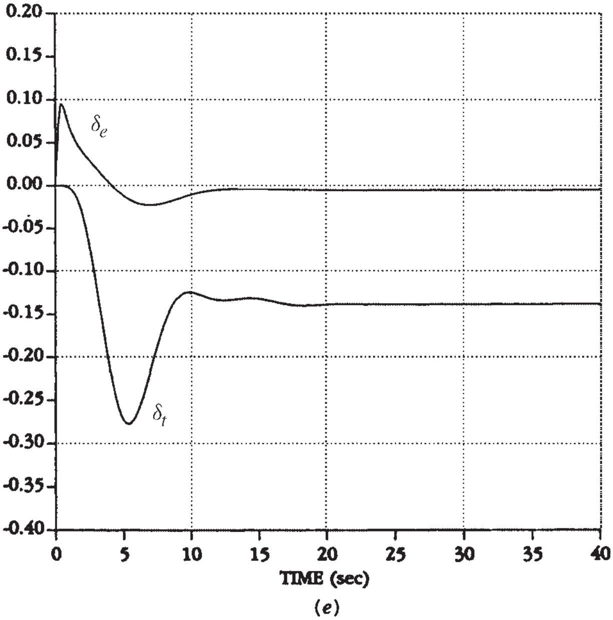


Figure 5.5-9 (*continued*)

It should be mentioned that determining an initial stabilizing gain K_0 for program LQ is not easy. In this example, we used the root-locus techniques from Chapter 4 to find the initial gain. Other approaches were discussed earlier in the subsection entitled "Constrained Feedback Matrix." ■

5.6 MODEL-FOLLOWING DESIGN

In Section 4.3 we discussed flying qualities and gave the military flying qualities specifications for the various aircraft modes. These desirable flying qualities could be viewed as constituting an *ideal model with good performance* which we would like to reproduce in the actual aircraft. For instance, to obtain good longitudinal performance we could select suitable short-period and phugoid poles from the flying qualities specifications tabulated in Section 4.3. Then we could determine a state-variable realization of an ideal model with this behavior (see Stern and Henke, 1971). Finally, we could design a control system to make the actual aircraft behave like this ideal model.

This approach to control system design is the powerful model-following design technique. In this section we show how to design controllers that make the aircraft behave like a desired model. We will discuss two fundamentally different sorts of model-following control, "explicit" and "implicit," which result in controllers of different structure (Armstrong, 1980; Kreindler and Rothschild, 1976; O'Brien and Broussard, 1978).

Explicit Model-Following Control

Regulation with Model-Following Behavior First, we will consider the regulator problem, where the objective is to drive the plant state to zero. Then we will treat the more difficult tracker or servo problem, where the plant is to follow a reference command with behavior like the prescribed model. Let the plant be described in state-variable form by

$$\dot{x} = Ax + Bu \tag{5.6-1}$$

$$y = Cx \tag{5.6-2}$$

$$z = Hx, \tag{5.6-3}$$

with state $x(t) \in \mathbf{R}^n$ and control input $u(t) \in \mathbf{R}^m$. The measured output $y(t)$ is available for feedback purposes.

A model is prescribed with dynamics

$$\underline{\dot{x}} = \underline{Ax} \tag{5.6-4}$$

$$\underline{z} = \underline{Hx}, \tag{5.6-5}$$

where the model matrix $\underline{A}$ reflects a system with desirable handling qualities such as speed of response, overshoot, and so on. The model states suitable for feedback purposes are given by

$$\underline{y} = \underline{Cx} \tag{5.6-6}$$

Model quantities will be denoted by underbars or the subscript m.

Notice that the model has no reference input, since we are considering the regulator problem here. That is, the plant should have the same unforced response as the model, which translates into suitable locations of the poles.

It is desired to select the plant control $u(t)$ so that the plant performance output $z(t)$ matches the model output $\underline{z}(t)$, for then the plant will exhibit the desirable time response of the model. That is, we should like to minimize the *model mismatch error*

$$e = \underline{z} - z = \underline{Hx} - Hx \tag{5.6-7}$$

To achieve this control objective, let us select the performance index

$$J = \tfrac{1}{2}\int_0^\infty (e^{\mathrm{T}}Qe + u^{\mathrm{T}}Ru)\,dt, \tag{5.6-8}$$

with $Q > 0$ (to ensure that all components of the error vanish) and $R > 0$.

We can cast this model-matching problem into the form of the regulator problem whose solution appears in Table 5.3-1 as follows.

Define the augmented state $x' = [x^{\mathrm{T}} \quad \underline{x}^{\mathrm{T}}]^{\mathrm{T}}$ and the augmented system

$$\dot{x}' \begin{bmatrix} A & 0 \\ 0 & \underline{A} \end{bmatrix} x' + \begin{bmatrix} B \\ 0 \end{bmatrix} u \equiv A'x' + B'u \tag{5.6-9}$$

$$y' = \begin{bmatrix} y \\ \underline{y} \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & \underline{C} \end{bmatrix} x' \equiv C'x', \tag{5.6-10}$$

so that

$$e = \begin{bmatrix} -H & \underline{H} \end{bmatrix} x' \equiv H'x' \tag{5.6-11}$$

Then the PI (5.6-8) may be written

$$J = \tfrac{1}{2} \int_0^\infty ((x')^{\mathrm{T}} Q' x' + u^{\mathrm{T}} R u)\, dt, \tag{5.6-12}$$

with

$$Q' = \begin{bmatrix} H^{\mathrm{T}} Q H & -H^{\mathrm{T}} Q \underline{H} \\ -\underline{H}^{\mathrm{T}} Q H & \underline{H}^{\mathrm{T}} Q \underline{H} \end{bmatrix} \tag{5.6-13}$$

At this point it is clear that the design equations of Table 5.3-1 can be used to select $u(t)$ if the primed quantities A', B', C', Q' are used there. The conditions for convergence of the algorithm in Table 5.3-2 require that (A', B', C') be output stabilizable and $(\sqrt{Q'}, A')$ be detectable. Since the model matrix $\underline{A}$ is certainly stable, the block-diagonal form of A' and C' shows that output stabilizability of the plant (A, B, C) is required. The second condition requires detectability of the plant (H, A).

The form of the resulting output feedback control law is quite interesting. Indeed, the optimal feedback is of the form

$$u = -K'y' \equiv -\begin{bmatrix} K_p & K_m \end{bmatrix} y' = -K_p y - K_m \underline{y} \tag{5.6-14}$$

Thus, not only the plant output but also the *model* output is required. That is, the model acts as a *compensator* to drive the plant states to zero in such a fashion that the performance output $z(t)$ follows the model output $\underline{z}(t)$.

Tracking with Model-Following Behavior Unfortunately, while the model-following regulator problem has a direct solution that is easy to obtain, the model-following *tracker* problem is not so easy. In this situation, we should like the plant (5.6-1) to (5.6-3) to behave like the model

$$\dot{\underline{x}} = \underline{Ax} + \underline{B}r \tag{5.6-15}$$

$$\underline{z} = \underline{Hx}, \tag{5.6-16}$$

which is *driven by the reference input* $r(t)$. The approach above yields

$$\dot{x}' = \begin{bmatrix} A & 0 \\ 0 & \underline{A} \end{bmatrix} x' + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ B \end{bmatrix} r \equiv A'x' + B'u + G'r, \tag{5.6-17}$$

and thus the derivation in Section 5.3 results in a PI that contains a term in $r(t)$, for which the determination of the optimal feedback gains is not easy (Lewis, 1986).

A convenient technique for designing a practical tracker is the *command generator tracker (CGT)* technique, where the tracking problem is converted into a regulator problem (Franklin et al., 1986). In this approach, a generator system is assumed for the reference input. We will apply it here.

Thus, suppose that for some initial conditions the reference command $r(t)$ satisfies the differential equation

$$r^{(d)} + a_1 r^{(d-1)} + \cdots + a_d r = 0 \tag{5.6-18}$$

for a given degree d and set of coefficients a_i. Most command signals of interest satisfy such an equation. For instance, the unit step of magnitude r_0 satisfies

$$\dot{r} = 0, \tag{5.6-19}$$

with $r(0) = r_0$, while the ramp (velocity command) with slope v_0 satisfies

$$\ddot{r} = 0, \tag{5.6-20}$$

with $r(0) = 0$, $r(0) = v_0$. We call (5.6-18) the *command generator system.*

Define the command generator characteristic polynomial as

$$\Delta(s) = s^d + a_1 s^{d-1} + \cdots + a_d \tag{5.6-21}$$

Then denoting d/dt in the time domain by D, we may write

$$\Delta(D)r = 0 \tag{5.6-22}$$

Multiplying the augmented dynamics (5.6-17) by $\Delta(D)$ results in the modified system

$$\dot{\xi} = A'\xi + B'\mu, \tag{5.6-23}$$

where the modified state and control input are

$$\xi = \Delta(D)x' = (x')^{(d)} + a_1(x')^{(d-1)} + \cdots + a_d x' \tag{5.6-24}$$

$$\mu = \Delta(D)u = u^{(d)} + a_1 u^{(d-1)} + \cdots + a_d u \tag{5.6-25}$$

The reason for these manipulations is that because of (5.6-22), the reference command $r(t)$ does not appear in (5.6-23). Let us partition ξ as

$$\xi = \begin{bmatrix} \xi_p \\ \hline \xi_m \end{bmatrix} \tag{5.6-26}$$

Applying $\Delta(D)$ to the model mismatch error (5.6-7) results in

$$\Delta(D)e = \begin{bmatrix} -H & \underline{H} \end{bmatrix} \xi = H'\xi \tag{5.6-27}$$

This may be expressed in terms of state variables using the observability canonical form (Kailath, 1980), which for scalar $e(t)$ and $d = 3$, for instance, is

$$\dot{\epsilon} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} \epsilon + \begin{bmatrix} 0 \\ H' \end{bmatrix} \xi \equiv F\epsilon + \begin{bmatrix} 0 \\ H' \end{bmatrix} \xi \tag{5.6-28}$$

$$e = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \epsilon, \tag{5.6-29}$$

where $\epsilon(t) = \begin{bmatrix} e & \dot{e} & \cdots & e^{(d-1)} \end{bmatrix}^{\mathrm{T}}$ is the vector of the error and its first $d - 1$ derivatives.

Collecting all the dynamics (5.6-23) to (5.6-28) into one system yields

$$\frac{d}{dt}\begin{bmatrix} \epsilon \\ \xi \end{bmatrix} = \left[\begin{array}{c|c} F & \begin{matrix} 0 \\ H' \end{matrix} \\ \hline 0 & A' \end{array}\right] \begin{bmatrix} \epsilon \\ \xi \end{bmatrix} + \begin{bmatrix} 0 \\ \hline B' \end{bmatrix} \mu \tag{5.6-30}$$

Let us now note what we have achieved. Using the command generator polynomial $\Delta(s)$, we have prefiltered the augmented state, control input, and error to obtain a system (5.6-30) *that is not driven by the reference input* $r(t)$. Using this system we may now perform an LQ *regulator* design, since if its state goes to zero, the tracking error $e(t)$ vanishes. That is, by performing a regulator design (using Table 5.3-1) for (5.6-30), we may design a *tracker* control system that causes the original plant to follow the reference command with performance like that of the ideal model.

For the regulator design, we will take the outputs available for feedback in (5.6-30) as

$$v = \begin{bmatrix} I & 0 & 0 \\ 0 & C & 0 \\ 0 & 0 & \underline{C} \end{bmatrix} \begin{bmatrix} \epsilon \\ \xi_p \\ \xi_m \end{bmatrix} \tag{5.6-31}$$

To achieve small error without using too much control energy, we may select the PI (5.6-8) (with $u(t)$ replaced by $\mu(t)$). According to (5.6-29), the error is given in terms of the state of (5.6-30) by

$$e = h \begin{bmatrix} \epsilon \\ \xi \end{bmatrix}, \tag{5.6-32}$$

with $h = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$ the first row of the identity matrix. Therefore, in the PI we should weight the state of (5.6-30) using

$$Q' = h^{\mathrm{T}} Q h \tag{5.6-33}$$

Since the observability canonical form is always observable, the augmented system (5.6-30) is detectable if the plant (H, A) and the model $(\underline{H}, \underline{A})$ are both detectable.

Now, by applying the equations of Table 5.3-1 to the system (5.6-30) with outputs (5.6-31) and PI weights Q' and R, we may compute the control gains in the control law

$$\mu = -\begin{bmatrix} K_\epsilon & K_p & K_m \end{bmatrix} \begin{bmatrix} \epsilon \\ C\xi_p \\ \underline{C}\xi_m \end{bmatrix} \tag{5.6-34}$$

or

$$\Delta(s)u = -K_\epsilon \epsilon - K_p C \Delta(s) x - K_m \underline{C} \Delta(s) \underline{x} \tag{5.6-35}$$

To determine the optimal tracking control input $u(t)$ for the original system, write this as

$$\Delta(s)(u + K_p y + K_m \underline{y}) = -K_\epsilon \epsilon \equiv -\begin{bmatrix} K_d & \cdots & K_2 & K_1 \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \\ \vdots \\ e^{(d-1)} \end{bmatrix} \tag{5.6-36}$$

Thus, we obtain the transfer function

$$\frac{u + K_P y + K_m \underline{y}}{e} = -\frac{K_1 s^{d-1} + \cdots + K_{d-1} s + K_d}{s^d + a_1 s^{d-1} + \cdots + a_d}, \tag{5.6-37}$$

which may be implemented in reachability canonical form (Kailath, 1980) to obtain the control structure shown in Figure 5.6-1.

CGT Structure The structure of this *model-following command generator tracker (CGT)* is very interesting. It consists of an output feedback K_p, a feedforward compensator that is nothing but the reference model with a gain of K_m, and an additional feedforward filter in the error channel that guarantees perfect tracking. Note that if $d = 1$ so that $r(t)$ is a unit step, the error filter is a PI controller. If $d = 2$ so that $r(t)$

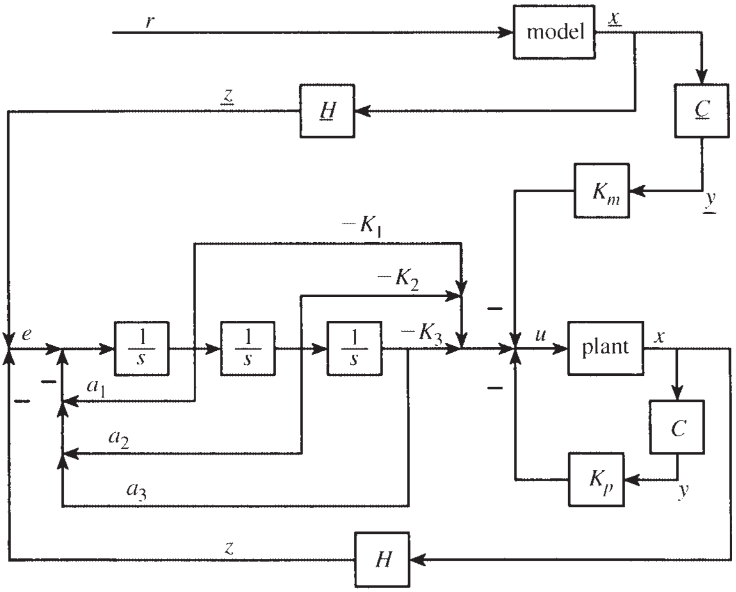


Figure 5.6-1 Explicit model-following command generator tracker for $d = 3$.

is a ramp, the error filter consists of two integrators, resulting in a type-2 system that gives zero steady-state error. What this means is the CGT design *automatically adds the compensator of appropriate structure to guarantee that the system has the correct type for perfect tracking*.

It is extremely interesting to note that the augmented state description (5.6-30) is nothing but the state description of Figure 5.6-1. It should be emphasized that this technique is extremely direct to apply. Indeed, given the prescribed model and the command generator polynomial $\Delta(s)$, the system (5.6-30)/(5.6-31) may be written down immediately and the equations in Table 5.3-1 used to select the feedback gains.

A word on the command generator assumption (5.6-22) is in order. In point of fact, for aircraft applications $r(t)$ is usually the pilot's command input. For control system design it is not necessary to determine the actual coefficients a_i that describe the pilot command, although this is one approach (Kreindler and Rothschild, 1976). Instead, the performance objectives should be taken into account to select $\Delta(s)$. For instance, if it desired for the aircraft to follow a position command, we may select the command generator $\dot{r} = 0$. On the other hand, if the aircraft should follow a rate (velocity) command, we may select $\ddot{r} = 0$. Then when the actual command input $r(t)$ is applied (which may be neither a unit step nor a unit ramp), the aircraft will exhibit the appropriate closed-loop behavior.

Implicit Model-Following Control

We will now discuss a formulation that results in a radically different sort of control scheme. In explicit model following, which is also called *model in the system control*, the model explicitly appeared in the controller as a feedforward compensator. On the other hand, implicit model following, also called *model in the performance index*, is a completely different approach in which the model does not appear in the control structure. Indeed, implicit model following can be viewed simply as a technique for selecting the weighting matrices in the PI in a meaningful way [see Armstrong (1980) and Kreindler and Rothschild (1976)].

Suppose that the performance output $z(t)$ of the plant prescribed by (5.6-1) to (5.6-3) is required to follow the model given by

$$\dot{\underline{z}} = \underline{A}\,\underline{z} \tag{5.6-38}$$

The model matrix $\underline{A}$ has poles corresponding to desirable handling qualities of the plant, such as may be found in *Mil. Spec. 1797* (1987) and Stern and Henke (1971).

When the control objective is met, the performance output will satisfy the differential equation (5.6-38). Thus, we may define an error by

$$e = \dot{z} - \underline{A}z \tag{5.6-39}$$

This is a different sort of error than we have seen before.

To make $e(t)$ small without using too much control energy, we may choose $u(t)$ to minimize the PI (5.6-8). Since $\dot{z} = HAx + HBu$, this becomes

$$J = \tfrac{1}{2}\int_0^\infty [(HAx + HBu - \underline{A}Hx)^{\mathrm{T}} Q (HAx + HBu - \underline{A}Hx) + u^{\mathrm{T}} Ru]dt \tag{5.6-40}$$

or

$$J = \tfrac{1}{2}\int_0^\infty (x^{\mathrm{T}} Q' x + 2x^{\mathrm{T}} Wu + u^{\mathrm{T}} R' u)dt, \tag{5.6-41}$$

where

$$\begin{aligned} Q' &= (HA - \underline{A}H)^{\mathrm{T}} Q (HA - \underline{A}H) \\ W &= (HA - \underline{A}H)^{\mathrm{T}} QHB, \quad R' = (B^T H^T QHB + R) \end{aligned} \tag{5.6-42}$$

The additional term in W is a *cross-weighting* between $u(t)$ and $x(t)$.

In Table 5.3-1 we have given the LQ regulator design equations to determine the optimal output feedback gains for the case $W = 0$. By using techniques like those in that derivation (see the problems), we may derive the modified design equations for the case of $W \neq 0$. They are

$$0 = A_c^{\mathrm{T}} P + PA_c + Q + C^{\mathrm{T}} K^{\mathrm{T}} RKC - W\ \ KC - C^{\mathrm{T}} K^{\mathrm{T}} W^{\mathrm{T}} \tag{5.6-43}$$

$$0 = A_c S + SA_c^{\mathrm{T}} + X \tag{5.6-44}$$

$$0 = RKCSC^{\mathrm{T}} - (PB + W)^{\mathrm{T}} SC^{\mathrm{T}}, \tag{5.6-45}$$

where

$$A_c = A - BKC \tag{5.6-46}$$

The optimal cost is still given by

$$J - \tfrac{1}{2}\mathrm{tr}(PX) \tag{5.6-47}$$

To find the optimal output feedback gains in

$$u = -Ky \tag{5.6-48}$$

for implicit model following, it is only necessary to solve these design equations using Q', W, and R'. For this, a technique like that in Table 5.3-2 may be used. Alternatively, algorithms such as the Simplex or Davidon-Fletcher-Powell may be employed.

Note that implicit model following in the regulator case is nothing but a convenient technique for selecting the PI weighting matrices Q', R' (and W) to guarantee desirable behavior, since the right-hand sides of (5.6-42) are known. Indeed, it is reasonable to select $R = \rho I$ and $Q = I$.

It is possible to design a tracking control system using implicit model following by using the CGT approach. However, this system has an undesirable structure from the point of view of aircraft control since it generally requires derivatives of the performance output $z(t)$.

Example 5.6-1: Automatic Flare Control by Model-Following Design. Model-following design may be used to design a control system that makes the aircraft behave like an ideal model (Kreindler and Rothschild, 1976). Such a model may be constructed using the military flying qualities requirements discussed in Section 4.3 so that it has good performance. However, this is not the only use for model-following design in aircraft controls.

In this example, we complete the design of the automatic landing system that was begun in Example 5.5-5. There, we constructed a glide-slope coupler whose function is to conduct an aircraft down a glide path toward the runway. Here we will show that explicit model-following design may be used to design the automatic flare control system whose function is to cause the aircraft to flare gently to a touchdown.

(a) *Determining the Reference Model.* The control system is basically an altitude-hold system with a time-varying reference or commanded altitude $\underline{h}t$. A gentle flare is described by an exponential, so that the commanded altitude should obey the differential equation

$$\dot{\underline{h}} = -\frac{1}{\tau}\underline{h} + r, \quad \text{initial condition } \underline{h}(0) = \underline{h}_0, \tag{1}$$

where τ and $\underline{h}_0$ are chosen for the desired flare characteristics. Equation (1) is the reference model (see 5.6-15)

$$\begin{aligned} \dot{\underline{x}} &= \underline{Ax} + \underline{B}r \\ \underline{z} &= \underline{Hx}, \end{aligned} \tag{2}$$

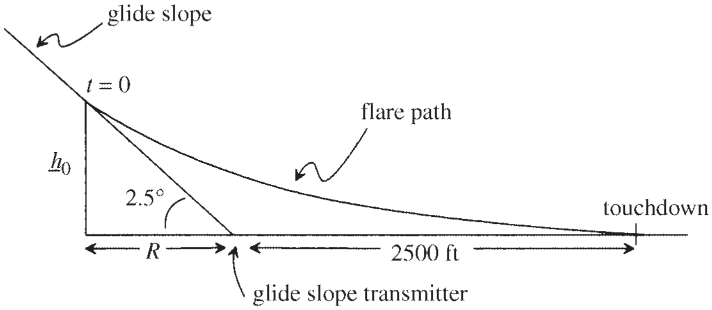


Figure 5.6-2 Flare-path geometry.

with $\underline{A} = -1/\tau$, $\underline{B} = 1$, $\underline{H} = 1$. Thus, $\underline{z} = \underline{x} = \underline{h}$. The model reference input is $r(t)$, which is equal to the constant value of zero in this example. Then

$$\underline{h}(t) = \underline{h}_0 e^{-t/\tau} \tag{3}$$

The geometry of the flare path shown in Figure 5.6-2 may be used to determine the flare time constant τ and initial altitude $\underline{h}_0$ (see Blakelock, 1965). In Example 5.5-5 we designed a glide-slope coupler for a total velocity of $V_T = 250\text{ft}/s$. Thus, on the glide path the rate of descent is

$$\dot{h} = V_t \sin(-2.5^\circ) \approx -V_T \frac{2.5}{57.2958} = -10.91\,\text{ft/s} \tag{4}$$

The flare control system is turned on at time $t = 0$ shown in the figure. Therefore, for (1) we obtain $\dot{\underline{h}}(0) = -10.91$, and

$$\underline{h}_0 = -\tau\dot{\underline{h}}(0) = 10.91\,\tau\,\text{ft} \tag{5}$$

The distance R is thus given by

$$R = \frac{\underline{h}_0}{\tan(2.5^\circ)} \approx \underline{h}_0 \times \frac{57.2958}{2.5} = V_T\tau\,\text{ft} \tag{6}$$

If it is desired to touch down 2500 ft beyond the glide-slope transmitter, and if we assume that $\underline{h}(t)$ given in (1) vanishes in 4τ seconds, then

$$4\tau V_T = R + 2500 = V_T\tau + 2500 \tag{7}$$

or

$$3\tau V_T = 2500, \tag{8}$$

so that

$$\tau = 3.333\,\text{s} \tag{9}$$

This yields the reference model

$$\dot{\underline{h}} = -0.3\underline{h} + r, \quad \underline{h}_0 = 36.37 \text{ ft}, \tag{10}$$

with reference input $r(t)$ taking the constant value of zero.

(b) *Basic Aircraft and Controller.* The flare control system is shown in Figure 5.6-3. For small flight-path angles the aircraft altitude is given by

$$\dot{h} = V_T \sin\gamma = V_T \sin(\theta - \alpha) \approx V_T\theta - V_T\alpha, \tag{11}$$

which is the same as the equation in Example 5.5-5 for $\dot{d}$ (with d the off-glide path distance) without the term in γ_r. What this means is that an altitude-hold system is given by the lower d-hold channel in the glide-slope coupler in Figure 5.5-8, with d replaced everywhere by h and γ_r removed. Indeed, the control gains in that system were used as an initial stabilizing guess in the LQ design for this example.

In this example we want to illustrate the model-following design procedure for the h control channel only. A complete design would include a second velocity-hold channel exactly as in the glide-scope coupler.

We used low-velocity longitudinal Jacobians for a medium-sized transport linearized about $V_T = 250$ ft/s, cg $= 0.25\bar{c}$, as in Example 5.5-5. For the flare control system h channel, we may use the short-period approximation, with, however, θ retained due to the need to compute the altitude using (11).

The state of the aircraft plus the lead compensator is

$$x = [\alpha \quad \theta \quad q \quad h \quad \delta_e \quad x_1]^{\mathrm{T}} \tag{12}$$

with x_1 the compensator state (see Figure 5.6-3). The performance output is

$$z = h = [0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0]x = Hx \tag{13}$$

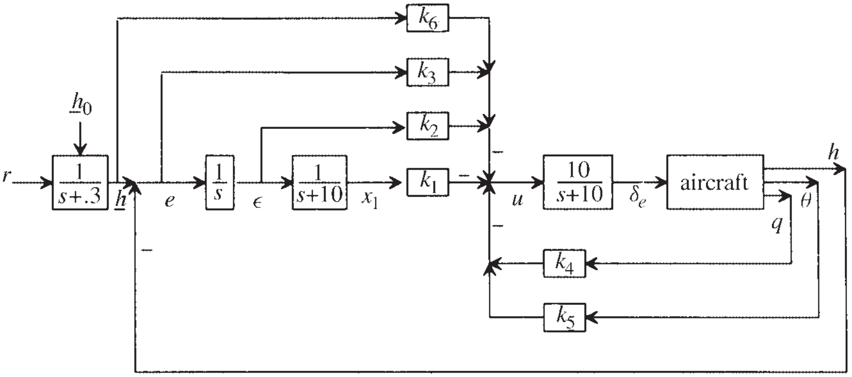


Figure 5.6-3 Automatic flare control system.

and the control input $u(t)$ is the elevator servo command. According to the figure, the measured outputs corresponding to the aircraft and the lead compensator are

$$y = [x_1 \quad q \quad \theta]^{\mathrm{T}} \tag{14}$$

(c) *Explicit Model-Following Control*. We should like the reference output $z(t)$ to follow the model altitude $\underline{h}(t)$ given by (2)/(10). Since the model's reference input $r(t)$ has the constant value of zero, $r(t)$ satisfies the differential equation

$$\dot{r} = 0, \tag{15}$$

so that the command generator polynomial (5.6-21) is given by

$$\Delta(s) = s \tag{16}$$

The model mismatch latitude error (5.6-7) is given by

$$e = \underline{h} - h \tag{17}$$

Therefore, the observability canonical form realization (5.6-28) is

$$\dot{\epsilon} = [-H \quad \underline{H}]\xi = [0 \quad 0 \quad 0 \quad -1 \quad 0 \quad 0 \quad | \quad 1]\xi, \tag{18}$$

with $\xi(t)$ the modified state $\Delta(s)[x^{\mathrm{T}} \quad \underline{x}]^{\mathrm{T}}$.

According to (18), $F = 0$ in the augmented system (5.6-30). Thus, we are required to incorporate an integrator in the control system [see (5.6-37) and Figure 5.6-1]. This we have already done in Figure 5.6-3.

The overall dynamics of the modified system (5.6-30) are given by

$$\begin{aligned} \dot{X} &= AX + Bu \\ y &= CX, \end{aligned} \tag{19}$$

with X the augmented state that contains the basic aircraft and compensator dynamics, the model dynamics (10), and the integrator required by (18). For convenience, we will order the states differently than in (5.6-30), taking

$$X = [\alpha \quad \theta \quad q \quad h \quad \delta_e \quad x_1 \quad \epsilon \quad \underline{h}]^{\mathrm{T}} \tag{20}$$

According to Figure 5.6-3, the outputs are

$$Y = [x_1 \quad \epsilon \quad e \quad q \quad \theta \quad \underline{h}]^{\mathrm{T}} \tag{21}$$

The model state $\underline{h}$ is included as an output due to the development leading to (5.6-37).

With this structure, the plant matrices are given by

$$A = \begin{bmatrix} -0.64627 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -0.7739 & 0 & -0.52977 & 0 & -0.011 & 0 & 0 & 0 \\ -250 & 250 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -10 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.3 \end{bmatrix} \tag{22}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 10 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{23}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

Then, according to Figure 5.6-3, the control input $u(t)$ is given by

$$u = -Ky = (k_1 x_1 + k_2 \epsilon + k_3 e + k_4 q + k_5 \theta + k_6 \underline{h}) \tag{25}$$

The control structure shown in Figure 5.6-3 and described here is nothing but the structure required for model following according to Figure 5.6-1.

(d) *PI and LQ Control Gain Design.* Although the explicit model-following design technique discussed in this section involves using the LQ *regulator* design equations from Table 5.3-1 on the augmented system (5.6-30), we have found that the results are generally better using LQ *tracker design with time-weighted PI*. Thus, we used the design equations in Table 5.5-1 with the auxiliary matrices

$$\begin{aligned} G &= [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]^{\mathrm{T}} \\ F &= [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^{\mathrm{T}} \\ H &= [0 \quad 0 \quad 0 \quad -1 \quad 0 \quad 0 \quad 0 \quad 1]^{\mathrm{T}}, \end{aligned} \tag{26}$$

which were determined from Figure 5.6-3. (Note the redefinition of the matrix H.)

The PI was selected as

$$J = \tfrac{1}{2} \int_0^\infty (q t^2 \tilde{e}^2 + \tilde{u}^2) dt \tag{27}$$

It is important to note that a sensible formulation of the problem has resulted in the appearance of *only one design parameter*, q, *in the PI*. Thus, we will not be faced with tuning many design parameters in an effort to obtain suitable responses. In view of the fact that there are eight states and six control gains to determine, this is quite significant. No steady-state error weighting is used in the PI since the plant is of type 1.

After several design iterations using the software of Appendix B to solve for K using the design equations in Table 5.5-1, we decided on $q = 0.001$ and obtained the control gains

$$K = [593.4 \quad -59.30 \quad 6.154 \quad -0.56 \quad -1.00 \quad -0.01852] \tag{28}$$

The closed-loop poles were at

$$\begin{aligned} -0.15 &\pm j0.23 \\ -0.15 &\pm j1.02 \\ -0.30&, -0.92 \\ -9.43&, -10.22 \end{aligned} \tag{29}$$

Note that the model pole of $s = -0.3$ has not moved since it is uncontrollable.

(e) *Simulation and Discussion*. The controlled flare is shown in Figure 5.6-4—it matches the desired flare $\underline{h}(t)$ very well. To obtain this graph it is necessary to use initial conditions $x(0)$ corresponding to the equilibrium state on the glide slope from Example 5.5-5. The flight-path angle γ is shown in Figure 5.6-5a. Shown in Figure 5.6-5b is the elevator command, δ_e; in examining this figure recall that upward elevator deflection (i.e., back stick) is defined as negative.

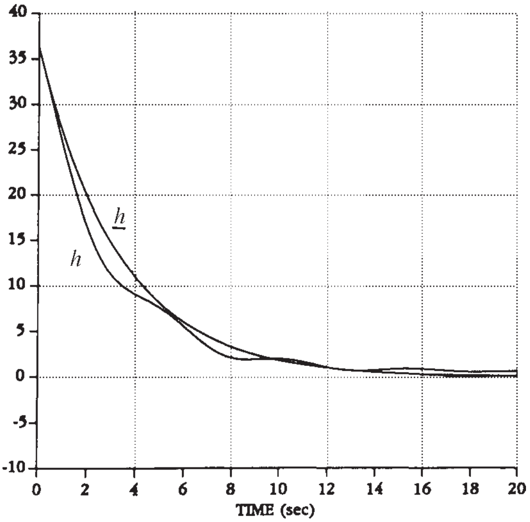


Figure 5.6-4 Controlled flare, altitude in feet.

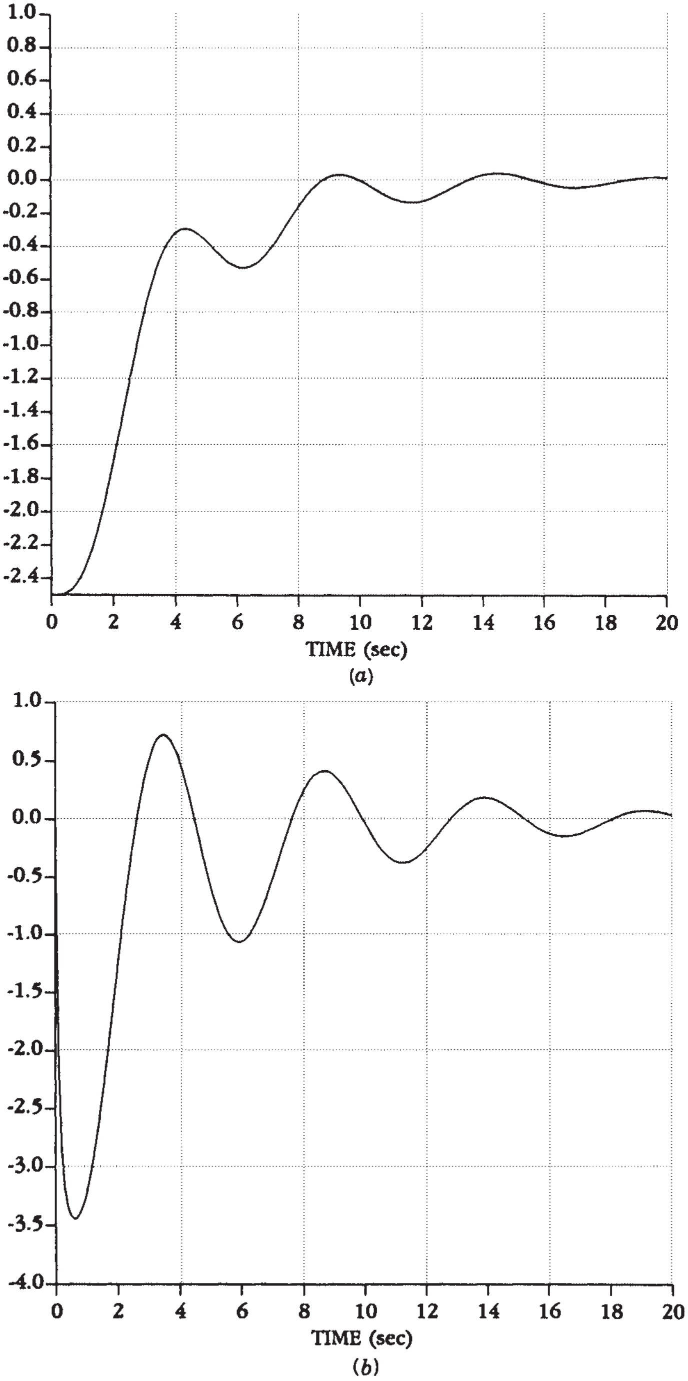


Figure 5.6-5 Aircraft response during controlled flare: (*a*) flight-path angle γ (deg); (*b*) elevator command δ_e (deg).

The poles in (29) are quite slow and there is one badly damped pair. However, the time responses are acceptable. This is because the flare control system is engaged with the aircraft on the glide path, so that there are no sudden reference command changes to excite the underdamped mode. Moreover, the flare is gentle, so that the time scale of the desired motion is on the order of the time scale of the closed-loop poles.

Although the control gain from the model state $\underline{h}$ to elevator servo command u is small, it plays a very important function. As may be seen in Blakelock (1965), the tendency of the flare control system without model state feedforward is to lag behind the desired response. This results in a flare that is always below the desired path and requires a modification in the design flare time constant τ. The feedforward of h corrects this problem in a simple manner.

Using the gains in (28), the compensator in the forward error channel of Figure 5.6-3 has the transfer function

$$\frac{k_1}{s(s+10)} + \frac{k_2}{s} + k_3 = \frac{6.154(s+0.364)}{s+10}, \tag{30}$$

where the pole at $s = 0$ has been canceled by a zero at $s = 0$ to yield a simplified compensator. Thus, there is no integrator in the feedforward path, and the model-following behavior does not rely on the system being of type 1. The ratio of the zero to the poles in the lead compensator is excessive, and the design may be repeated using, for instance, a compensator pole at $s = -5$ instead of $s = -10$ (and no integrator). ■

5.7 LINEAR QUADRATIC DESIGN WITH FULL STATE FEEDBACK

In the previous sections of this chapter we have seen how to design control systems using a variety of modern control techniques that rely only on measuring a system *output*. These output feedback approaches are very suitable for aircraft control design since they allow us to design a compensator with any desired dynamical structure. This cannot be accomplished using full state feedback.

In this section we intend to explore full state-variable feedback in the linear quadratic regulator (LQR) for the insight it provides. That is, for the system

$$\dot{x} = Ax + Bu, \tag{5.7-1}$$

with $x \in \mathbf{R}^n$, $u \in \mathbf{R}^m$, we want to examine control laws of the form

$$u = -Kx, \tag{5.7-2}$$

which result in the closed-loop system

$$\dot{x} = (A - BK)x \equiv A_c x \tag{5.7-3}$$

In the previous sections we defined the measurable output

$$y = Cx \tag{5.7-4}$$

and restricted ourselves to controls of the form

$$u = -Ky = -KCx \tag{5.7-5}$$

Here we plan to examine the simplifications in the control design equations that come about when $C = I$. As we will see, we can draw some conclusions that will give more insight into modern control theory.

The Relevance of State Feedback

Although all the states are seldom measurable in aircraft control systems, we have several objectives for looking at state-variable feedback design in this section. First, it is clear that state feedback is just the special case of output feedback with $C = I$. That is, it assumes that all the states can be measured. Thus, the theory for state-variable feedback will tell us *the best performance that we can expect* in the closed-loop system by using static output feedback, where all of the states are not available as measurements.

Second, the output feedback design equations in Tables 5.3-1, 5.4-1, and 5.5-1 are not the LQR equations with which the reader may be familiar. We would like to show how they relate to the more traditional Riccati equation.

If all the states are involved in the feedback, there are some very powerful stability results of which the reader should be aware. Indeed, under some reasonable assumptions it is possible to *guarantee the stability of the closed-loop system* using the optimal LQ state feedback gain. Similar theoretical results for output feedback have not yet been discovered.

Finally, we will need state feedback in Chapter 6 when we discuss dynamic regulators and LQR/LTR robust design. A limitation of state feedback is that all the states are not generally available, but only the outputs are measured. However, we can design a full state feedback $u = -Kx$ and then a dynamic observer to estimate the states from the measured outputs. Then the state *estimates* $\hat{x}$ may be fed back, instead of the states themselves, in a control law such as $u = -K\hat{x}$. The combination of state feedback plus an observer is called a *dynamic regulator*. It is a compensator of the sort used in classical control, but it is easy to design for multivariable systems, overcoming a deficiency of the classical approach, where multiloop and MIMO systems are hard to deal with.

The Riccati Equation and Kalman Gain

By setting $C = I$ all of our work in Sections 5.3 and 5.4 applies to state feedback. That is, all the work of deriving the control design equations for state feedback has already been done. Let us see how the LQR design equations simplify in the case of full state feedback.

To regulate the performance output

$$z = Hx \tag{5.7-6}$$

to zero, let us select the PI

$$J = \tfrac{1}{2}\int_0^{\infty}(x^{\mathrm{T}}Qx + u^{\mathrm{T}}Ru)dt, \tag{5.7-7}$$

with $Q = H^{\mathrm{T}}H \geq 0, R > 0$.

The output feedback gain K in (5.7-5) that minimizes the PI may be found using the design equations in Table 5.3-1. To obtain the optimal state feedback in (5.7-2), we may simply set $C = I$ in the table. The results are

$$0 = A_c^{\mathrm{T}}P + PA_c + Q + K^{\mathrm{T}}RK \tag{5.7-8}$$

$$0 = A_CS + SA_c^{\mathrm{T}} + X \tag{5.7-9}$$

$$K = R^{-1}B^{\mathrm{T}}PSS^{-1}, \tag{5.7-10}$$

where the initial state autocorrelation is

$$X = E\{x(0)x^{\mathrm{T}}(0)\} \tag{5.7-11}$$

The problems in computing the output feedback gains include the need to know X and the selection of an initial stabilizing gain K_0 for the algorithm in Table 5.3-2. Moreover, although we gave conditions for the convergence to a local minimum of the algorithm in that table, little is known about the necessary and sufficient conditions for the existence of an output feedback gain that satisfies the design equations and stabilizes the plant.

All of these problems vanish in the case of state feedback, as we will now show. According to (5.7-10),

$$K = R^{-1}B^{\mathrm{T}}P, \tag{5.7-12}$$

that is, the solution S to (5.7-9) *is not needed to solve for the optimal state feedback gain*. The gain K is called the *Kalman gain*. Using (5.7-12) in (5.7-8) yields

$$0 = A_c^{\mathrm{T}}P + PA_c + Q + PBR^{-1}B^{\mathrm{T}}P \tag{5.7-13}$$

or, according to (5.7-3),

$$\begin{aligned} 0 &= (A - BR^{-1}B^{\mathrm{T}}P)^{\mathrm{T}}P + P(A - BR^{-1}B^{\mathrm{T}}P) \\ &\quad + Q + PBR^{-1}B^{\mathrm{T}}P \\ 0 &= A^{\mathrm{T}}P + PA + Q - PBR^{-1}B^{\mathrm{T}}P \end{aligned} \tag{5.7-14}$$

This matrix quadratic equation is called the *algebraic Riccati equation (ARE)*. It is named after Count J. F. Riccati, who used a related equation in the study of heat flow (Riccati, 1724). Since the equation is equal to its own transpose (verify!), the solution P is symmetric ($P = P^{\mathrm{T}}$).

Since S is not needed to find the optimal state feedback gain K, this gain does not depend on X in (5.7-9). That is, contrary to the case with output feedback, to compute the optimal state feedback gains *no information about the initial state* $x(0)$ *is needed*. Thus, it is not required to take expected values of the PI as we did in Section 5.3. Therefore, according to the development in that section, the optimal cost is given by

$$J = \tfrac{1}{2}x^{\mathrm{T}}(0)Px(0) \tag{5.7-15}$$

The state feedback LQR is summarized in Table 5.7-1.

Setting $C = I$ has allowed us to replace the solution of three coupled matrix equations by the solution of *one nonlinear matrix equation* for P. Then the Kalman gain is given in terms of P by (5.7-17). The importance of this is that there are many good techniques for solving the Riccati equation using *standard software packages* [e.g., ORACLS (Armstrong, 1980), MATRIX$_{\mathrm{x}}$ (1989), PC-MATLAB (Moler et al., 1987), and IMSL (1980)]. On the other hand, the specialized software for solving the output feedback problem in Tables 5.3-1, 5.4-1, or 5.5-1 can be used to solve the full state feedback problem by setting $C = I$.

Guaranteed Closed-Loop Stability

The theory for the LQ regulator with state feedback is well developed. In fact, the next stability result is so fundamental that we set it apart as a theorem (Lewis, 1986). The notion of detectability was introduced while discussing Table 5.3-2. We say that $(A,\ H)$ is detectable if there exists an L so that $A - LH$ is stable; this amounts to

TABLE 5.7-1 LQR with State Feedback

System Model

$$\dot{x} = Ax + Bu$$

Control

$$u = -Kx$$

Performance Index

$$J = \tfrac{1}{2}\int_0^\infty (x^{\mathrm{T}}Qx + u^{\mathrm{T}}Ru)dt$$

Optimal LQ Design Equations

- Algebraic Riccati equation (ARE)

$$0 = A^{\mathrm{T}}P + PA + Q - PBR^{-1}B^{\mathrm{T}}P \tag{5.7-16}$$

- Kalman gain

$$K = R^{-1}B^{\mathrm{T}}P \tag{5.7-17}$$

Optimal Cost

$$J = \tfrac{1}{2}x^{\mathrm{T}}(0)Px(0)$$

the observability of the unstable modes of A. We say that (A, B) is *stabilizable* if there exists a feedback gain K such that $A_c = A - BK$ is stable. This amounts to the controllability of the unstable modes of A.

Theorem. *Let H be any matrix so that $Q = H^T H$. Suppose that (H, A) is detectable. Then (A, B) is stabilizable if and only if:*

(a) There exists a unique positive-semidefinite solution P to the Riccati equation.

(b) The closed-loop system (5.7-3) is asymptotically stable if the Kalman gain K is computed using (5.7-17) in terms of this positive-semidefinite solution P. ■

This result is at the heart of modern control theory. Exactly as in classical control, it allows us to examine *open-loop* properties (i.e., detectability and stabilizability) and draw conclusions about the closed-loop system. As long as (H, A) is detectable, so that all the unstable modes appear in the PI, and (A, B) is stabilizable, so that the control $u(t)$ has sufficient influence on the system, the LQ regulator using state feedback will *guarantee* a stable closed-loop system. A similar easily understandable result has not yet been discovered for output feedback.

Detectability is implied by the stronger condition of observability, which is easy to check by verifying that the observability matrix has full rank n (see Section 5.3). Stabilizability is implied by controllability, which is easy to check by verifying that the controllability matrix has full rank n (see Section 5.2). Thus, the controllability of (A, B) and the observability of (H, A) guarantee closed-loop stability of the LQ regulator with state feedback.

This theorem, coupled with the availability of good software for solving the ARE, means that it is always straightforward to find a state-variable feedback gain K that stabilizes any stabilizable plant, no matter how many inputs or outputs it has.

Since output feedback amounts to a partial state feedback, it is clear that if the conditions of the theorem do not hold, we should not expect to be able to stabilize the plant using any output feedback (unless time-dependent weighting of the form t^k is used in the PI to avoid the observability requirement; see Section 5.5). Thus, in the case of output feedback design these conditions should hold *as a minimum*. In fact, we saw that the algorithm of Table 5.3-2 requires the detectability of $(\sqrt{Q}, A)$ and the output stabilizability of the system. Output stabilizability is a stronger condition than stabilizability.

In the case of a full state feedback, it is possible in simple examples to give a direct correlation between the PI weighting matrices and the closed-loop poles. Let us investigate this connection for systems obeying Newton's laws.

Example 5.7-1: LQR with State Feedback for Systems Obeying Newton's Laws In this example we will see that in the case of full state feedback for simple systems there is a direct connection between the PI weights and the closed-loop damping ratio and natural frequency.

Systems obeying Newton's laws may be described by the state equation

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u = Ax + Bu, \tag{1}$$

where the state is $x = [d \quad v]^T$ with $d(t)$ the position and $v(t)$ the velocity and the control $(u)t$ is an acceleration input. Indeed, note that (1) says nothing other than $\ddot{d} = u$, or $a = F/m$.

Let the PI be

$$J = \tfrac{1}{2}\int_0^{\infty} (x^{\mathrm{T}}Qx + u^2)\, dt, \tag{2}$$

with $Q = \mathrm{diag}\{q_d^2, qv\}$. In this example, we will see the effect of q_d and q_v. Note that it is not useful to include a separate control weighting r, since only the ratios q_d^2/r and q_v/r are important in J.

Since the Riccati solution P is symmetric, we may assume that

$$P = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} \tag{3}$$

for some scalars p_1, p_2, p_3 to be determined. Using A, B, Q, and $r = 1$ in the Riccati equation in Table 5.7-1 yields

$$\begin{aligned} 0 = &\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} + \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix}\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} q_d^2 & 0 \\ 1 & q_v \end{bmatrix} \\ &- \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix}\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} \end{aligned} \tag{4}$$

The reader should verify that this may be multiplied out to obtain the three scalar equations

$$0 = -p_2^2 + q_d^2 \tag{5a}$$

$$0 = p_1 - p_2 p_3 \tag{5b}$$

$$0 = 2p_2 - p_3^2 + q_v \tag{5c}$$

Solving these equations in the order (5a), (5c), (5b) gives

$$p_2 = q_d \tag{6a}$$

$$p_3 = \sqrt{2}\sqrt{q_d + \frac{qv}{2}} \tag{6b}$$

$$p_1 = q_d\sqrt{2}\sqrt{qd + \frac{q_v}{2}}, \tag{6c}$$

where we have selected the signs that make P positive definite.

According to Table 5.7-1, the Kalman gain is equal to

$$K + R^{-1}B^{\mathrm{T}}P = [0 \quad 1]\begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} = [p_2 \quad p_3] \tag{7}$$

Therefore,

$$K = \begin{bmatrix} q_d & \sqrt{2}\sqrt{q_d + \frac{q_v}{2}} \end{bmatrix} \tag{8}$$

It should be emphasized that in the case of state feedback we have been able to find an *explicit expression* for K in terms of the PI weights. This is not possible for output feedback.

Using (8), the closed-loop system matrix is found to be

$$A_c = (A - BK) = \begin{bmatrix} 0 & 1 \\ -q_d & -\sqrt{2}\sqrt{q_d + \frac{q_v}{2}} \end{bmatrix} \tag{9}$$

Therefore, the closed-loop characteristic polynomial is

$$\Delta_c(s) = |sI - A_c| = s^2 + 2\zeta\omega s + \omega^2, \tag{10}$$

with the optimal natural frequency ω and damping ratio ζ given by

$$\omega = \sqrt{q_d}, \quad \zeta = \frac{1}{\sqrt{2}}\sqrt{1 + \frac{q_v}{2q_d}} \tag{11}$$

It is now clear how selection of the weights in the PI affects the closed-loop behavior. Note that if no velocity weighting q_v is used, the damping ratio becomes the familiar $1/\sqrt{2}$.

Note that (A, B) is reachable since

$$U = [B \quad AB] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{12}$$

is nonsingular. The observability matrix is

$$0 = \begin{bmatrix} \sqrt{Q} \\ \sqrt{QA} \end{bmatrix} = \begin{bmatrix} q_d & 0 \\ 0 & \sqrt{d_v} \\ 0 & q_d \\ 0 & 0 \end{bmatrix} \tag{13}$$

Therefore, observability is guaranteed if and only if the position weighting q_d is greater than zero. Then the theorem says that we should be able to rely on a stable closed-loop system. Examining (11) makes it clear that this is indeed the case. ■

5.8 DYNAMIC INVERSION DESIGN

In this chapter we have presented some basic tools of modern control design for linear systems. Since aircraft are inherently nonlinear systems, applying these linear design tools means that one must design several linear controllers and then gain schedule them over the operating regime of the aircraft (see Problem 5.4-9). There are alternative techniques that can deal directly with the known nonlinearities of the aircraft dynamics using these nonlinearities in the controller to improve the system performance. These techniques are generally based on the *feedback* linearization approach (Slotine and Li, 1991) developed by Hunt et al. (1983) and Jacubczyk and Respondek (1980).

In this section we introduce the technique known as *dynamic inversion*, which has grown popular in recent years (Adams and Banda, 1993; Lane and Stengel, 1988; Enns et al., 1994; Tomlin et al., 1995; Wright Laboratory Report, 1996). The dynamic inversion controller takes into account the nonlinearities of the aircraft and thus does not require gain scheduling. As such it is suitable for a wide range of operating conditions, including high-angle-of-attack and hypervelocity design. To simulate the dynamic inversion control scheme, we use the technique of computer simulation for nonlinear dynamical systems given in Section 3.3. For this, we use the MATLAB software (*MATLAB Reference Guide*, 1994).

Though dynamics inversion is used for nonlinear systems and shows its true power there, we will start this section with a linear derivation and design example to get a feel for how it works. Then, we will study dynamic inversion controls design and simulation for a nonlinear aircraft.

Dynamic Inversion for Linear Systems

Derivation of Dynamic Inversion Controller Let the plant be described in state-variable form by

$$\dot{x} = Ax + Bu \tag{5.8-1}$$

$$y = Cx, \tag{5.8-2}$$

with state $x(t) \in \mathbf{R}^n$, control input $u(t) \in \mathbf{R}^m$, and output $y(t) \in \mathbf{R}^p$. The entire state $x(t)$ is available for feedback purposes.

It is assumed that the system is *square*, that is, the number of inputs m is equal to the number of outputs p so that vectors $u(t)$ and $y(t)$ have the same dimension. This often occurs for aircraft systems, since there is often one control actuator per degree of freedom. If this is not the case, we may make some amendments to the following procedure. For instance, in modern high-performance aircraft, there may be more actuators than degrees of freedom (e.g., elevators, horizontal stabilators, and thrust vectoring for longitudinal dynamics). In this event, the control dimension may be reduced to obtain a square system by several techniques, including ganging, pseudo control, and daisy chaining (Wright Laboratory Report, 1996). These are all techniques for *allocating control effectiveness* among several redundant actuators.

For a square system, then, it is desired to control the output $y(t)$ so that it follows a desired reference trajectory $r(t)$. Define the tracking error as

$$e(t) = r(t) - y(t) \tag{5.8-3}$$

In dynamic inversion, one differentiates the output $y(t)$ until the control $u(t)$ appears in the expression for the derivative. This is known technically as *input-output feedback linearization* (Slotine and Li, 1991). Taking the first derivative yields

$$\dot{y} = C\dot{x} = CAx + CBu, \tag{5.8-4}$$

where $u(t)$ appears if matrix CB is not zero. In this case, since the system is square, so is matrix CB. If CB is nonsingular, then we are done. If $CB = 0$, then we continue to differentiate, obtaining

$$\ddot{y} = C\dot{x} = CA\dot{x} + CB\dot{u} = CA^2x + CABu \tag{5.8-5}$$

If matrix CAB is nonsingular, we are done. If $CAB = 0$, we differentiate again, continuing until the coefficient multiplying $u(t)$ is nonzero.

For aircraft, it is generally the case that CB is nonsingular. This is because of the way in which the control actuators enter into the aircraft dynamics equations, with one actuator for each degree of freedom. Then, we may stop at (5.8-4). Define an auxiliary input $v(t)$ by

$$v = CBu + CAx - \dot{r} \tag{5.8-6}$$

so that

$$u = (CB)^{-1}(\dot{r} - CAx + v) \tag{5.8-7}$$

Substituting this expression for $u(t)$ into (5.8-4) yields

$$\begin{aligned} \dot{y} &= CAx + CB[(CB)^{-1}(\dot{r} - CAx + v)] \\ &= CAx + \dot{r} - CAx + v \end{aligned} \tag{5.8-8}$$

or

$$\dot{e} = -v \tag{5.8-9}$$

The auxiliary input $v(t)$ was selected to make expression (5.8-7) hold in order to cancel the term CAx and so that CB does not appear in (5.8-9).

System (5.8-9) is the *error dynamics*. To complete the design, it is only necessary to select $v(t)$ so that this system is stable. Due to the way in which $v(t)$ was defined by (5.8-7), the error dynamics have a very simple form; indeed, the error system has p poles at $s = 0$. This means that it is very easy to select $v(t)$ to stabilize this system. A variety of techniques may be used, including robust control, LQR/LTR (see Chapter 6), and other linear system design techniques (Adams and Banda, 1993; Lane and Stengel, 1988; Enns et al., 1994; Tomlin et al., 1995; Wright Laboratory Report, 1996).

A simple choice for $v(t)$ is

$$v = Ke \tag{5.8-10}$$

Then, one has the closed-loop error dynamics given by

$$\dot{e} = -Ke, \tag{5.8-11}$$

which is a stable system as long as gain matrix K is positive definite. In practice, one usually selects K diagonal to keep the control channels in the outer loop decoupled. The gain K should be selected so that the closed-loop system satisfied MILSPEC flying qualities requirements.

The overall dynamic inversion control input is given by

$$u = (CB)^{-1}(\dot{r} + Ke - CAx) \tag{5.8-12}$$

The control scheme given by this is shown in Figure 5.8-1. Note that (5.8-10) is simply an *outer proportional feedback tracking loop*, while (5.8-7) is an *inner control loop* using full state-variable feedback. This inner loop is called the *feedback linearization loop*. Its function is to make the system from $v(t)$ to $y(t)$ appear like a linear system with poles at the origin (5.8-9). This greatly simplifies the design of the outer tracking loop. There is also a feedforward term involving $\dot{r}(t)$, which is known as *velocity feedforward*. This greatly improves the tracking accuracy of the closed-loop system.

Note that to implement the dynamic inversion control algorithm (5.8-12) one must know CA and CB. That is, *a model of the aircraft dynamics is actually built into the controller*. This is what makes the outer control loop design so simple. Moreover, full state feedback is required for the inner loop.

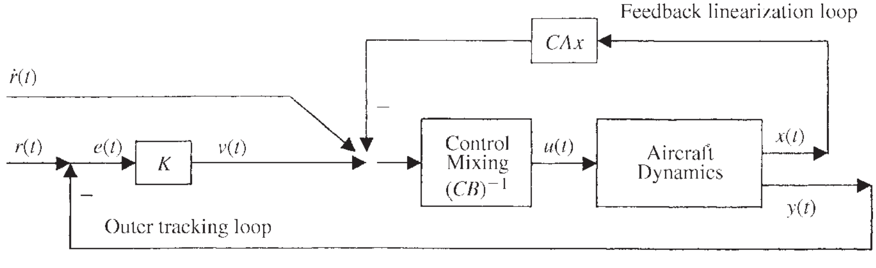


Figure 5.8-1 Dynamic inversion controller.

This completes the design of the dynamic inversion controller. The full power of this approach will be seen in the next subsection when we apply the technique to nonlinear aircraft systems.

Zero Dynamics Equation (5.8-11) only gives the error dynamics of the output $y(t)$. The full closed-loop system is obtained by substituting the control (5.8-7) into the state equation (5.8-1). This yields

$$\dot{x} = Ax + B(CB)^{-1}(\dot{r} - CAx + v)$$
$$\dot{x} = [I - B(CB)^{-1}]Ax + B(CB)^{-1}(\dot{r} + v) \tag{5.8-13}$$

The *zero dynamics* are defined as the dynamics of the system when the input $v(t)$ is selected to give an output $y(t)$ equal to zero. Since $y(t) = 0$, then $\dot{y}(t) = 0$, so that (5.8-9) shows that

$$v = -\dot{e} = \dot{y} - \dot{r} = -\dot{r}$$

Substituting this value for $v(t)$ into (5.8-13) yields the zero dynamics

$$\dot{x} = [I - B(CB)^{-1}C]Ax \equiv A_Z x \tag{5.8-14}$$

Note that the dimension of the entire state is n, while the dimension of the error dynamics (5.8-11) is $p < n$. The error dynamics are guaranteed stable by the choice of $v(t)$; however, there remain *n-p* poles that may or may not be stable. These poles are unobservable selecting the output $y(t)$, and so they cannot be moved using the dynamic inversion controller. These *n-p* poles are exactly given by the zero dynamics. If some of these internal zeros are NMP, then the closed-loop system designed by dynamic inversion will be unstable.

The poles of the matrix A_z consist of p poles at the origin [namely, the poles of error dynamics system (5.8-9)] plus the *n-p* internal zeros. Define the operator

$$P \equiv I - B(CB)^{-1}C \tag{5.8-15}$$

Note that $P^2 = P$ so that P is a projection. Furthermore, $PB \equiv [I - B(CB)^{-1}C]\ B = 0$ and $CP \equiv C[I - B(CB)^{-1}C] = 0$, so that P is the projection on the null space of C along the range of B. Thus, $A_z = PA$ describes those dynamics that are both in the null space of C and in range perpendicular of B. These are precisely the modes that are unobservable using the output $y(t)$ and cannot be controlled using the dynamic inversion approach.

Selection of Controlled Variables For dynamic inversion to be successful, it is necessary to *select the controlled variable* (*CV*) $y(t)$ *so that the zero dynamics are stable*.

This may be checked by computing A_z for the selected output matrix C and finding its poles. If they are not stable, then a new C-matrix must be selected. Once a suitable C-matrix has been found, the p poles of the error dynamics may be selected using (5.8-10).

The outputs to be controlled in fighter aircraft are usually selected as:

$$\begin{aligned}&\text{pitch axis CV:} && q + n_{zp}/V_{CO}\\ &\text{roll axis CV:} && p + \alpha r\\ &\text{yaw axis CV:} && r - \alpha p - (g \sin\phi \cos\theta)/V + k\beta\end{aligned} \tag{5.8-16}$$

Discussion of these controlled variables may be found in the work of Enns et al. (1994). These CVs are suitable for most conventional flight regimes and piloting tasks. They may need modifications for high-α or very-low-speed flight.

The pitch-axis CV is motivated by the C-star criterion C^*. See the discussion in Section 4.3 on C^*. One uses the normal acceleration at the pilot's station n_{zp} and not n_z at the center of gravity of the aircraft since the latter yields unstable zero dynamics. The crossover velocity V_{CO} of the CV should be selected to match the MILSPEC requirements on n_{zp} and pitch rate q. Even using this CV, dynamic inversion design can destabilize the phugoid mode. This problem may be avoided by adding a small airspeed term to the pitch-axis CV (Enns et al., 1994) to obtain

$$\text{pitch axis CV: } q + n_{zp}/V_{CO} + kv_T \tag{5.8-17}$$

The gain k is selected so that the zero dynamics are stable. This is illustrated in the next example. The roll- and yaw-axis Cvs generally do not have such a stability problem.

In the next example we show how to select CVs and design a dynamic inversion controller for linearized longitudinal dynamics. To verify the performance of the controller, we employ MATLAB using the technique for computer simulation of systems that was introduced in Section 3.3. This technique applies for linear or nonlinear systems and employs directly the actual controller. Once the simulation results are satisfactory, this controller can simply be cut out of the code and programmed into the aircraft computer.

Example 5.8-1: Dynamic Inversion Design for Linear F-16 Longitudinal Dynamics Consider the linearized F-16 longitudinal dynamics of Chapter 3. Including an elevator actuator, the states are given by

$$x = [v_T \quad \alpha \quad \theta \quad q \quad \delta_e]^T \tag{1}$$

The control input is the elevator input u_e. The A-and B-matrices are given by

$$A = \begin{bmatrix} -0.1270 & -235.0000 & -32.2000 & -9.5100 & -0.2440 \\ 0 & -0.9690 & 0 & 0.9080 & -0.0020 \\ 0 & 0 & 0 & 1.0000 & 0 \\ 0 & -4.5600 & 0 & -1.5800 & -0.2000 \\ 0 & 0 & 0 & 0 & -20.2000 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 20.2 \end{bmatrix} \tag{2}$$

In Chapter 4 both the normal acceleration n_z at the cg and the normal acceleration n_{zp} at the pilot's station are given. One has

$$n_z = [0.004 \quad 15.88 \quad 0 \quad 1.481 \quad 0.33]x \tag{3}$$

$$n_{zp} = [0.004 \quad 16.2620 \quad 0 \quad 0.9780 \quad -0.0485]x \tag{4}$$

Computing $C^* = n_{zp} + 12.4q$ as in Chapter 4 yields

$$C^* = [0.004 \quad 16.2620 \quad 0 \quad 13.3780 \quad -0.0485]x \tag{5}$$

(a) *Zero Dynamics for Different Controlled Variables*. To find a suitable CV, one may compute the zero dynamics $A_z = [I - B(CB)^{-1}C]A$ for the above outputs. Then, the eigenvalues of A_z may be determined. One notes that all these computations are very easy using the MATLAB software.

For $y(t) = n_z$ one obtains

$$13.8346, -65717, -0.1242, 0.0633, 0$$

Since the number of inputs and outputs is $p = 1$, one obtains one pole at zero, corresponding to the error dynamics (5.8-9). The remaining poles of A_z are the zero dynamics. Since there are unstable zeros, performing dynamic inversion design using n_z as the CV would destabilize the system.

For $y(t) = n_{zp}$ one obtains the poles of A_z as

$$-3.6652 \pm 7.6280i, \quad -0.1244, \quad 0.0550, \quad 0$$

One has the pole at zero of the error dynamics, plus the internal zeros using this choice for $y(t)$. The situation is better, and there is only one slightly unstable pole in the zero dynamics.

For $y(t) = C^*$ one obtains

$$-56.3022, \quad -2.1389, \quad -0.1252, \quad 0.0325, \quad 0,$$

which still reveals a slightly unstable zero dynamics pole.

Accordingly, none of these would be appropriate choices for the CV for dynamic inversion design, though n_{zp} and C^* are both better than n_z. The trouble is the velocity term in n_{zp} and C^*. One may correct the problem by adding a small airspeed term to n_{zp} or C^* (Enns et al., 1994). After a few tries, we decided on

$$y(t) = C^* - 0.014v_T = n_{zp} + 12.4q - 0.014v_T, \tag{6}$$

which yields

$$y(t) = [-.01 \quad 16.2620 \quad 0 \quad 13.3780 \quad -0.0485]x = Cx \tag{7}$$

Computing now A_z and finding its eigenvalues yield

$$-56.2243, \quad -2.0209, \quad -0.1423, \quad -0.0758, \quad 0$$

The unstable zero dynamics pole is gone and so we proceed to simulate the dynamic inversion controller.

(b) *Simulation of the Dynamic Inversion Controller.* To simulate the dynamic inversion controller given in Figure 5.8-1, we use a method of computer simulation that is based on the nonlinear state equation $\dot{x} = f(x, u)$. This technique was detailed in Section 3.3 and works for either linear or nonlinear systems. MATLAB has a built-in function that performs Runge-Kutta integration and is very convenient to use here.

The third-order Runge-Kutta integrator in MATLAB is called *ode23*, and it requires a MATLAB M file containing the nonlinear state dynamics. This M file is given in Figure 5.8-2. The form of this M file is very important. Note that one first computes the dynamic inverse controller (5.8-12), and then computes the state equation derivatives for the aircraft. Thus, the first portion of the M file is exactly the code required to implement the controller on the actual aircraft.

It is important to note that the dynamic inversion controller must know the aircraft dynamics *A,B,C*. That is, the controller must contain a model of the aircraft dynamics.

We selected the desired trajectory $r(t)$ equal to a unit step, since if the step response of the controlled aircraft is suitable, then the controller has a good performance for a wide range of pilot input commands.

The MATLAB command lines required to run the simulation are given by

```
»[t,x]=ode23('F16LinDynInv',[0 5],[0 0 0 0 0]');
»y=x*cstarmodd';
»plot(t,y)
```

```
% Inverse Dynamics Controller for F16 Linear Dynamics

function xdot=Fl6LinDynInv(t,x)
global y
%      VT= x(1);      ! True airspeed
%      ALPH= x(2);    ! Angle of Attack in rads.
%      THTA= x(3);    ! Pitch attitude in rads.
%      Q   = x(4);    ! Pitch rate rad/s
%      elev= x(5);    ! elevator actuator
% Inverse Dynamics Controller

    % Model of aircraft

    a=   [ −0.1270  −235.0000   −32.2000   −9.5100   −0.2440
             0    −0.9690           0       0.9080   −0.0020
               0            0            0         1.0000       0
               0    −4.5600              0       −1.5800   −0.2000
               0           0             0           0      −20.2000  ];
    b=    [0 0 0 0 20.20]’;
    c=    [ −0.01   16.2620          0   13.3780    −0.0485];
   &  % y=cstar’ modifed

    % command input

    r= 1   ;  % check step response
    rdot= 0 ;

    % controller parameters

    K= 10;

    %  plant outputs, tracking errors, and control inputs

    y= c*x ;   % y= cstar’ modified
    e= r-y ;
    v= K*e ;
    w= rdot - c*a*x + v ;
    u= inv(c*b) * w ;
    tht1=0
    uelev= u ;

% Aircraft State Equations

xdot(1)= −0.1270*x(1) −235.0*x(2) −32.3*x(3) −9.51*x(4) −0.244*x(5) +62.8&thtl ;
xdot(2)=             −0.9690*x(2)          +0.908*x(4) −0.002*x(5) −0.04*thtl ;
xdot(3)=                                          x(4)  ;
xdot(4)=             −4.56*x(2)             −1.58*x(4) −0.2*x(5)   ;
xdot(5)=                                               −020.2*x(5) +20.2*uelev;

xdot=xdot’ ;
```

Figure 5.8-2 Dynamic inversion controller and simulation code.

where cstarmodd is the C-matrix defined by (7). The second argument in *ode23* specifies the integration time interval 0 to 5 s, and the third argument specifies zero initial conditions $x(0)$.

The output $y(t)$ is shown in Figure 5.8-3. If the time constant is not suitable according to MILSPEC requirements, one may simply select another value of K in (5.8-12) and repeat the simulation. The pitch rate is shown in Figure 5.8-4.

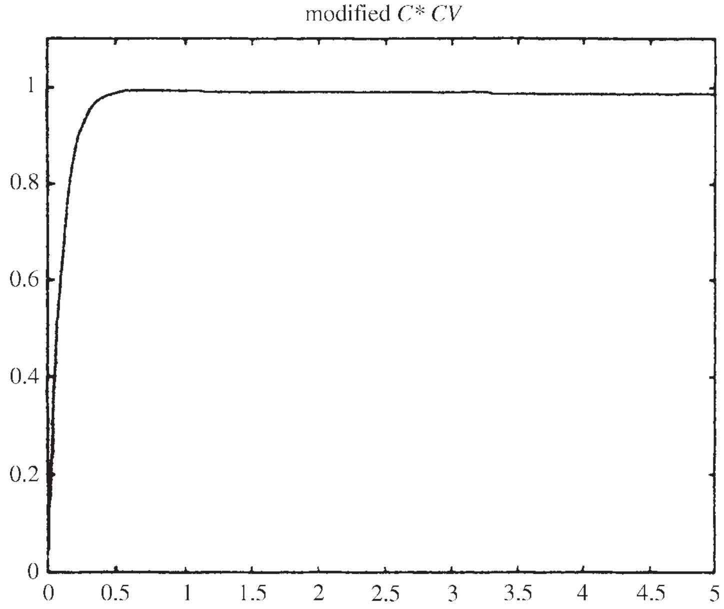


Figure 5.8-3 Modified C^* controlled variable.

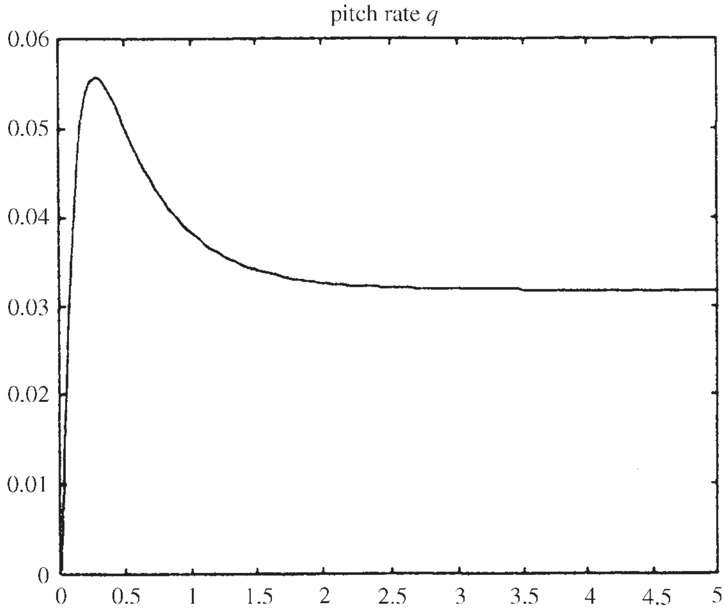


Figure 5.8-4 Pitch rate $q(t)$.

■

A Pathological Case In aircraft control one generally has CB nonsingular in (5.8-4). If $CB = 0$ one may proceed as discussed there. However, in pathological situations it may occur that CB is neither zero nor nonsingular. Then one must proceed as follows.

Differentiating repeatedly one obtains

$$\ddot{y} = C\ddot{x} = CA\dot{x} + CB\dot{u} = CA^2x + CB\dot{u} + CABu = CA^2x + C[B \quad AB]\begin{bmatrix}\dot{u}\\ u\end{bmatrix}$$

$$\dddot{y} = C\dddot{x} = CA^3x + C[B \quad AB \quad A^2B]\begin{bmatrix}\ddot{u}\\ \dot{u}\\ u\end{bmatrix}$$

Continuing for n steps, with n the number of states, one obtains the nth derivative of $y(t)$ as

$$y^{(n)}(t) = CA^n x(t) + CU_n\underline{u}(t), \tag{5.8-18}$$

where the controllability matrix is

$$U_n = [B \quad AB \quad \cdots \quad A^{n-1}B] \tag{5.8-19}$$

and $\underline{u}(t)$ is a vector of $u(t)$ and its first $n-1$ derivatives.

Now, if the system is controllable, then U_n has rank n. If in addition the C-matrix has rank p, then CU_n has rank p. In this case, though CU_n is not square, it has a right inverse given by

$$(CU_n)^+ = (CU_n)^T[(CU_n)(CU_n)^T]^{-1} \tag{5.8-20}$$

for note that $(CU_n)(CU_n)^+ = I$, the $p \times p$ identity matrix. Thus, one may define

$$\underline{u} = (CU_n)^+(r^{(n)} - CA^n x + v) \tag{5.8-21}$$

and substitute into (5.8-18) to obtain

$$y^{(n)}(t) = CA^n x(t) + CU_n\underline{u}(t) = CA^n x(t) + CU_n[(CU_n)^+(r^{(n)} - CA^n x + v)],$$

which yields

$$y^{(n)}(t) = r^{(n)} + v \tag{5.8-22}$$

or

$$e^{(n)} = -v \tag{5.8-23}$$

This is the error dynamics. It has p poles at the origin.

Selecting now the outer loop structure given by

$$v = K_{n-1}e^{(n-1)} + \cdots + K_0 e \tag{5.8-24}$$

gives the closed-loop error dynamics

$$e^{(n)} + K_{n-1}e^{(n-1)} + \cdots + K_0 e = 0 \tag{5.8-25}$$

The gains K_i can be selected to make this system stable. Note that this requires feedforward of the tracking error $e(t)$ and its derivatives.

In this pathological case, the inverse dynamics controller is given by (5.8-21) and (5.8-24). A dynamical system can then be employed to extract the control input $u(t)$ from its derivative vector $\underline{u}(t)$.

Dynamic Inversion for Nonlinear Systems

Since the aircraft is inherently a nonlinear system, we will now discuss dynamic inversion control for nonlinear systems (Slotine and Li, 1991; Enns et al., 1994; Wright Laboratory Report, 1996). Dynamic inversion is one of few control techniques that can directly be extended to nonlinear systems.

Derivation of Dynamic Inversion Controller Let the plant be described in nonlinear state-variable form by

$$\dot{x} = f(x) + g(x)u \tag{5.8-26}$$

$$y = h(x), \tag{5.8-27}$$

with state $x(t) \in R^n$, control input $u(t) \in R^m$, and output $y(t) \in R^p$. The entire state $x(t)$ is available for feedback purposes. It is assumed that the system is *square*, that is, the number of inputs m is equal to the number of outputs p so that vectors $u(t)$ and $y(t)$ have the same dimension.

Note that the system is linear in the control input $u(t)$. This generally holds for aircraft systems, though if it does not and one has instead the more general state equation

$$\dot{x} = f(x, u),$$

one can use a modified form of the upcoming development (Enns et al., 1994).

To make the system follow a desired trajectory $r(t)$, the tracking error is defined as

$$e(t) = r(t) - y(t) \tag{5.8-28}$$

Differentiate the output to obtain

$$\dot{y} = \frac{\partial h}{\partial x}\dot{x} = \frac{\partial h}{\partial x}f(x) + \frac{\partial h}{\partial x}g(x)u \equiv F(x) + G(x)u \tag{5.8-29}$$

Define now the control input by

$$u = G^{-1}(x)[-F(x) + \dot{r} + v] \tag{5.8-30}$$

with $v(t)$ an auxiliary input to be defined. Substituting this expression into (5.8-29) yields the error dynamics

$$\dot{e} = -v \tag{5.8-31}$$

Now, any linear design technique, including robust control techniques, LQR/LTR, and so on, can be used to select $v(t)$ to stabilize this linear system with p poles at the origin. One convenient choice is simply

$$v = Ke \tag{5.8-32}$$

with K positive definite.

The overall dynamic inversion controller is given by

$$u = G^{-1}(x)[-F(x) + \dot{r} + Ke] \tag{5.8-33}$$

This controller is depicted in Figure 5.8-5. It requires full state feedback for the inner loop.

Note that the control $u(t)$ has been selected to make the plant from $v(t)$ to $y(t)$ be simply a linear system with p poles at the origin. This is accomplished by the inner feedback linearization loop, which is now nonlinear. Then, an outer tracking loop is closed to complete the design. Any linear design technique, including robust control, H-infinity, or LQG/LTR, may be used for this outer-loop design.

It is important to note that the control (5.8-33) contains a model of the aircraft dynamics, since it requires $F(x)$ and $G(x)$. Therefore, to implement it, one must know the nonlinear functions in the aircraft equation. In the upcoming example, this amounts to including the nonlinear aircraft functions in the controller, but in practice it usually entails including *full lookup tables* in the controller. This can become cumbersome but is possible with today's computing systems.

In aircraft systems, $G(x)$ is usually nonsingular. If $G(x)$ is not nonsingular, then one must take more steps to derive the controller. If $G(x) = 0$, one may proceed as in Slotine and Li (1991), repeatedly differentiating $y(t)$ using Lie derivatives.

The CVs are selected as detailed for linear systems, though nonlinear versions of the controlled outputs may be used. For instance, for the roll axis one might use the nonlinear version $p \cos \alpha + r \sin \alpha$ (Enns et al., 1994).

In the nonlinear case, it is more difficult to test the selected controlled variables than in the linear case, since one does not have the artifice of the zero dynamic matrix A_z. However, one may linearize the nonlinearities and use a version of the technique presented for linear systems. Specifically, the full closed-loop dynamics are given by

$$\begin{aligned} \dot{x} &= f(x) + g(x)G^{-1}(x)[-F(x) + \dot{r} + v] \\ &= \left[I - gG^{-1}\frac{\partial h}{\partial x}\right]f(x) + gG^{-1}[\dot{r} + v] \end{aligned} \tag{5.8-34}$$

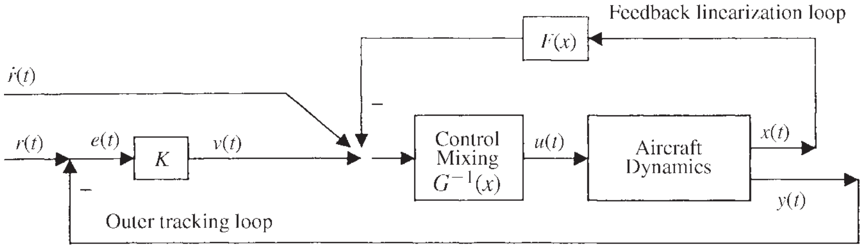


Figure 5.8-5 Nonlinear dynamic inversion controller.

The zero dynamics are given by

$$\dot{x} = \left[I - gG^{-1}\frac{\partial h}{\partial x}\right]f(x) \tag{5.8-35}$$

These may be linearized to determine the suitability of the CV at a specific operating point. Stability may also be checked by simulation. Simply simulate (5.8-35) in MATLAB, selecting different initial conditions and verifying that the state converges to zero in each case. This amounts to plotting a phase portrait of the zero dynamics (Tomlin et al., 1995).

Example 5.8-2: Dynamic Inversion Design for Nonlinear Longitudinal Dynamics We now present a nonlinear version of Example 5.8-1. A longitudinal model $\dot{x} = f(x) + g(x)u$ of an aircraft similar to that resented in Chapter 3 is given in Figure 5.8-6. The states are

$$x = [v_T \quad \alpha \quad \theta \quad q \quad \delta_e]^T \tag{1}$$

and the control input is the elevator actuator input u_e.

The normal acceleration is given by

$$n_z = \overline{q}S(C_L \cos\alpha + C_D \sin\alpha)/mg \tag{2}$$

and the normal acceleration at the pilot's station is

$$n_{zp} = n_z + 15\dot{q}/g = \overline{q}S(C_L \cos\alpha + C_D \sin\alpha)/mg + 15M/gI_{yy}, \tag{3}$$

with M the pitching moment (MOM in Figure 5.8-6). The output is selected as

$$y = C^* = n_{zp} + 12.4q \equiv h(x) \tag{4}$$

We aim to apply (5.8-33) to compute the dynamic inversion controller. To do this, we must determine

$$F(x) = \frac{\partial h}{\partial x}f(x) \quad \text{and} \quad G(x) = \frac{\partial h}{\partial x}g(x) \tag{5}$$

$f(x)$ and $g(x)$ are easily determined from the nonlinear dynamics and are given in Figure 5.8-7. Finding $\partial h/\partial x$ is tedious and the results are as follows. First, $\partial n_z/\partial x$ is given as

$$\begin{aligned}
\frac{\partial n_z}{\partial v_T} &= \rho v_T S(C_L \cos\alpha + C_D \sin\alpha)/mg \\
\frac{\partial n_z}{\partial \alpha} &= \overline{q}S[(C_D + 4.58)\cos\alpha - 0.515C_L \sin\alpha]/mg \\
\frac{\partial n_z}{\partial \theta} &= 0, \quad \frac{\partial n_z}{\partial q} = 0, \quad \frac{\partial n_z}{\partial \delta_e} = 0
\end{aligned} \tag{6}$$

```
% Nonlinear Longitudinal Aircraft Model (for small airplane)
% B. Stevens file modified by F. Lewis on 8 May 2000

function xdot=NonLinDynInv(t,x);

% Definition of some constants for the aircraft used for simulation;
    WEIGHT = 2300.0; G=32.2; MASS=WEIGHT/G;
    IYY = 2094.;
    RHO = 2.377E-3;
    S = 175.0;
    CBAR = 4.89;
    CMQ = -12.0;
    RTOD = 57.29578;      ! radians to degrees

      VT= x(1);        ! True airspeed
      ALPH= x(2);      ! Angle of Attack in rads.
      THTA= x(3);      ! Pitch attitude in rads
      Q  = x(4);       ! Pitch rate rad/s
      EL= x(5);        ! elevator actuator

% Computed control inputs are thtl (throttle) and uelev (elev. act. command)

      GAM= THTA - ALPH;
      CBV= 0.5*CBAR/VT;
      CL = 0.25 + 4.58*ALPH;              ! Linear lift curve
      CM = 0.015 - 0.75*ALPH - 0.9*x(5);  ! Linear pitching moment
      CD = .038 + .053*CL*CL;             ! Parabolic drag
      QBAR= 0.5*RHO*VT*VT;                ! Dynamic pressure
      LIFT= QBAR*S*CL;
      DRAG= QBAR*S*CD;
      MOM = QBAR*S*CBAR*(CM + CBV*CMQ*Q); ! Added pitch damping
      FT= (338.02 + 1.5651*vt - .00884*vt**2 )* thtl; ! Nonlinear Thrust

% State Equations

      xdot(1)= (FT*cos(ALPH) - DRAG - WEIGHT*sin(GAM) )/MASS;
      xdot(2)= (-FT*sin(ALPH) - LIFT + WEIGHT*cos(GAM))/(MASS*VT)   + Q;
      xdot(3)= Q;
      xdot(4)= MOM/IYY;
      xdot(5)= -20.2*x(5) + 20.2*uelev;

% outputs

      nz= QBAR*S*(CL*cos(ALPH) + CD*sin(ALPH)) / (G*MASS) ! Normal accel.
      nzp= nz + 15*MOM/(g*IYY) ! Normal accel. at pilot's station
      cstar= nzp - 12.4q  !  controlled variable
```

Figure 5.8-6 Nonlinear model of aircraft longitudinal dynamics.

Next, one has $\partial M/\partial x$ given by

$$\begin{aligned}
\frac{\partial M}{\partial v_T} &= \rho v_T S\overline{c}C_M + \frac{S\overline{c}^2}{4}\rho q C_{mq} \\
\frac{\partial M}{\partial \alpha} &= -0.75\,\overline{q}\,\overline{c}S \\
\frac{\partial M}{\partial \theta} &= 0 \\
\frac{\partial M}{\partial q} &= \frac{S\overline{c}^2\overline{q}}{2v_T}C_{mq} \\
\frac{\partial M}{\partial \delta_e} &= -0.9\overline{q}\,\overline{c}S
\end{aligned} \tag{7}$$

Finally, one has

$$\frac{\partial C^*}{\partial x} = \frac{\partial n_z}{\partial x} + \frac{15}{gI_{yy}}\frac{\partial M}{\partial x} + k,$$

where $k = [0 \quad 0 \quad 0 \quad 12.40]^T$. All of these are included finally in Figure 5.8-7.

To simulate the dynamic inversion controller, one may write a single MATLAB M file containing both the controller in Figure 5.8-7 and the aircraft dynamics in Figure 5.8-6. The form of this M file will be similar in spirit to that used in Example 5.8-1. This is left for the enterprising reader.

It is very important to note that the dynamic inversion controller in Figure 5.8-7 requires full knowledge of all the nonlinear dynamics of the aircraft. In this example this entails including all the analytic expressions used in the aircraft model. However, in practice, it generally involves including full aircraft lookup tables in the controller. In this example, the Jacobian $\partial h/\partial x$ was computed analytically. In practice, one may use a numerical differentiation routine as part of the controller.

```
% Nonlinear Longitudinal Dynamic Inversion Controller
   function xdot=NonLinDynInvCtrlr(t,x);

% MODEL OF AIRCRAFT DYNAMICS USED IN CONTROLLER

% Definition of some constants for the aircraft used for simulation;
    WEIGHT = 2300.0; G=32.2; MASS=WEIGHT/G;
    IYY = 2094.;
    RHO = 2.377E-3;
    S = 175.0;
    CBAR = 4.89;
    CMQ = -12.0;
    RTOD = 57.29578;     % radians to degrees

      VT= x(1);      % True airspeed
      ALPH= x(2);    % Angle of Attack in rads.
      THTA= x(3);    % Pitch attitude in rads.
      Q   = x(4);    % Pitch rate rad/s
      EL= x(5);      % elevator actuator

% Computed control inputs are thtl (throttle) and EL (elevator command)
      GAM= THTA - ALPH;
      CBV= 0.5*CBAR/VT;
      CL = 0.25 + 4.58*ALPH;                 % Linear lift curve
      CM = 0.015 - 0.75*ALPH - 0.9*x(5);     % Linear pitching moment
      CD = .038 + 0.53*CL*CL;                % Parabolic drag
      QBAR= 0.5*RHO*VT*VT;                   % Dynamic pressure
      LIFT= QBAR*S*CL;
      DRAG= QBAR*S*CD;
      MOM = QBAR*S*CBAR*(CM + CBV*CMQ*Q); % Added pitch damping
      FT= (338.02 + 1.5651*vt - .00884*vt**2 )* thtl; % Nonlinear Thrust

% function f(x)
      f1= (FT*cos(ALPH) - DRAG - WEIGHT*sin(GAM) )/MASS;
      f2= (-FT*sin(ALPH) - LIFT + WEIGHT*cos(GAM))/(MASS*VT)  + Q;
      f3= Q;
      f4= MOM/IYY;
      f5= -20.2*x(5);

% function g(x)
            g=[0  0  0  0  20.2]' ;
```

Figure 5.8-7 Nonlinear dynamic inversion controller (Part I). Nonlinear dynamic inversion controller (Part II).

■

5.9 SUMMARY

In this chapter we showed how to use modern control techniques to design multivariable and multiloop aircraft flight control systems. The approach is based on the state-variable model and a mathematical performance criterion selected according to the performance objectives. The matrix of control gains is determined by solving explicit matrix equations using computer software. Using such an approach, all the feedback loops are closed simultaneously to yield the guaranteed performance desired. This is in contrast to the classical techniques of Chapter 4, which relied on trial-and-error successive loop closures to find the control gains individually.

Two basic modern design techniques were covered. In Section 5.2 we discussed eigenstructure assignment techniques that take advantage of the freedom inherent in design for systems with more than one input and/or output to assign the closed-loop poles and eigenvectors. In the remainder of the chapter we covered linear quadratic (LQ) techniques, where the control gains are selected to minimize generalized quadratic performance indices (PIs). Design equations were derived for the control gains minimizing these PIs and listed in tabular form for easy reference. The design equations may be solved for the control gains using software like that described in Appendix B.

In Section 5.5 the thrust was to introduce modified nonstandard PIs allowing LQ designs with only a small number of design parameters that require tuning for suitable performance. The point was made that successful control system design hinges on the selection of a suitable PI.

Our primary thrust was to use output feedback to allow the design of a compensator with any desired structure. The PI was an integral of the squares of the states and control inputs; thus the LQ techniques used in this chapter are *time-domain* techniques.

In Section 5.8 we discussed dynamic inversion design, which results in a controller with an inner feedback linearization loop and an outer tracking loop.

REFERENCES

Adams, R.J., and S. S. Banda. “Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis.” *IEEE Transactions on Control System Technology* 1, no. 2 (June 1993): 80–92.

AFWAL-TR-84-3008. “AFTI/F-16 Development and Integration Program, DFCS Phase Final Technical Report.” Fort Worth, Tex.: General Dynamics, December 1984.

Andry, A. N., Jr., E. Y. Shapiro, and J. C. Chung. “Eigenstructure Assignment for Linear Systems.” *IEEE Transactions on Aerospace and Electronic Systems* AES-19, no. 5 (September 1983): 711–729.

Andry, A. N., Jr., J. C. Chung, and E. Y. Shapiro. “Modalized Observers.” *IEEE Transactions on Automatic Control* AC-29, no. 7 (July 1984): 669–672.

Armstrong, E. S. *ORACLS: A Design System for Linear Multivariable Control*. New York: Marcel Dekker, 1980.

Bartels, R. H., and G. W. Stewart. "Solution of the Matrix Equation $AX + XB = C$." *Communications of the ACM* 15, no. 9 (September 1972): 820–826.

Blakelock, J. H. *Automatic Control of Aircraft and Missiles*. New York: Wiley, 1965.

Broussard, J., and N. Halyo. "Active Flutter Control using Discrete Optimal Constrained Dynamic Compensators." *Proceedings of the American Control Conference*, June 1983, pp. 1026–1034.

Bryson, A. E., Jr. "New Concepts in Control Theory, 1959–1984." *Journal of Guidance* 8, no. 4 (July–August 1985): 417–425.

Bryson, A. E., Jr., and Y.-C. Ho. *Applied Optimal Control*. New York: Hemisphere, 1975.

Davison, E. J., and I. J. Ferguson. "The Design of Controllers for the Multivariable Robust Servomechanism Problem Using Parameter Optimization Methods." *IEEE Transactions on Automatic Control* AC-26, no. 1 (February 1981): 93–110.

D'Azzo, J. J., and C. H. Houpis. *Linear Control System Analysis and Design*. New York: McGraw-Hill, 1988.

Doyle, J. C., and G. Stein. "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis." *IEEE Transactions on Automatic Control* AC-26, no. 1 (February 1981): 4–16.

Enns, D., D. Bugajski, R. Hendrick, and G. Stein. "Dynamic Inversion: An Evolving Methodology for Flight Control Design." *International Journal of Control* 59, no. 1 (1994): 71–91.

Franklin, G. F., J. D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Reading, Mass.: Addison-Wesley, 1986.

Gangsaas, D., K. R. Bruce, J. D. Blight, and U.-L. Ly. "Application of Modern Synthesis to Aircraft Control: Three Case Studies." *IEEE Transactions on Automatic Control* AC-31, no. 11 (November 1986): 995–1014.

Grimble, M. J., and M. A. Johnson. *Optimal Control and Stochastic Estimation: Theory and Applications*. Vol. 1. New York: Wiley, 1988.

Harvey, C. A., and G. Stein. "Quadratic Weights for Asymptotic Regulator Properties." *IEEE Transactions on Automatic Control* AC-23, no. 3 (1978): 378–387.

Hunt, L. R., R. Su, and G. Meyer. "Global Transformations of Nonlinear Systems." *IEEE Transactions on Automatic Control* 28 (1983): 24–31.

IMSL. *Library Contents Document*. 8th ed. Houston, Tex.: International Mathematical and Statistical Libraries, Inc., 1980.

Jacubczyk, B., and W. Respondek. "On Linearization of Control Systems." *Bulletin Academie Polonaise de Science et Mathematique* 28 (1980): 517–522.

Kailath, T. *Linear Systems*. Englewood Cliffs, N.J.: Prentice Hall, 1980.

Kalman, R. "Contributions to the Theory of Optimal Control." *Boletin de la Sociedad de Matematica Mexicana* 5 (1958): 102–119.

Kalman, R. E. "A New Approach to Linear Filtering and Prediction Problems." *Transactions of the ASME Journal of Basic Engineering* 82 (1960): 34–35.

Kreindler, E., and D. Rothschild. "Model-Following in Linear-Quadratic Optimization." *American Institute of Aeronautics and Astronautics Journal* 14, no. 7 (July 1976): 835–842.

Kwakernaak, H., and R. Sivan. *Linear Optimal Control Systems*. New York: Wiley, 1972.

Kwon, B.-H., and M.-J. Youn. "Eigenvalue-Generalized Eigenvector Assignment by Output Feedback." *IEEE Transactions on Automatic Control* AC-32, no. 5 (May 1987): 417–421.

Lane, S. H., and R. F. Stengel. "Flight Control Using Non-Linear Inverse Dynamics." *Automatica* 24, no. 4 (1988): 471–483.

Levine, W. S., and M. Athans. “On the Determination of the Optimal Constant Output Feedback Gains for Linear Multivariable Systems.” *IEEE Transactions on Automatic Control* AC-15, no. 1 (February 1970): 44–48.

Lewis, F. L. *Optimal Control*. New York: Wiley, 1986.

Ly, U.-L., A. E. Bryson, and R. H. Cannon. “Design of Low-Order Compensators Using Parameter Optimization.” *Automatica* 21, no. 3 (1985): 315–318.

MacFarlane, A.G.J. “The Calculation of Functionals of the Time and Frequency Response of a Linear Constant Coefficient Dynamical System.” *Quarterly Journal of Mechanical Applied Mathematics* 16, pt. 2 (1963): 259–271.

MATLAB Reference Guide. Natick, Mass.: The MathWorks, 1994.

MATRIX$_x$. Santa Clara, Calif.: Integrated Systems, Inc., 1989.

McRuer, D., I. Ashkenas, and D. Graham. *Aircraft Dynamics and Automatic Control*. Princeton N.J.: Princeton University Press, 1973.

Mil. Spec. 1797. “Flying Qualities of Piloted Vehicles.” 1987.

Moerder, D. D., and A. J. Calise. “Convergence of a Numerical Algorithm for Calculating Optimal Output Feedback Gains.” *IEEE Transactions on Automatic Control* AC-30, no. 9 (September 1985): 900–903.

Moler, C., J. Little, and S. Bangert. *PC-Matlab*. Sherborn, Mass.: The Mathworks, Inc., 1987.

Moore, Bruce C. “On the flexibility offered by state feedback in multivariable systems beyond closed loop eigenvalue assignment.” Decision and Control including the 14th Symposium on Adaptive Processes, 1975 IEEE Conference on. IEEE, 1975.

Nelder, J. A., and R. Mead. “A Simplex Method for Function Minimization.” *Computing Journal* 7 (1964): 308–313.

O’Brien, M. J., and J. R. Broussard. “Feedforward control to track the output of a forced model.” Decision and Control including the 17th Symposium on Adaptive Processes, 1978 IEEE Conference on. IEEE, 1979.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. New York: Cambridge University Press, 1986.

Quintana, V. H., M. A. Zohdy, and J. H. Anderson. “On the Design of Output Feedback Excitation Controllers of Synchronous Machines.” *IEEE Transactions of Power Apparatus Systems* PAS-95, no. 3 (1976): 954–961.

Riccati, J. F. “Animadversiones in aequationes differentiales secundi gradus.” *Actorum Eruditorum quae Lipsiae publicantur, Suppl*. 8 (1724): 66–73.

Shahian, B., and M. Hassul. *Control System Design Using MATLAB*. Englewood Cliffs, N.J.: Prentice-Hall, 1993.

Slotine, J.-J. E., and W. Li. *Applied Nonlinear Control*. Englewood Cliffs, N.J.: Prentice Hall, 1991.

Sobel, K., and E. Shapiro. “Eigenstructure assignment for design of multimode flight control systems.” IEEE Control Systems Magazine 5.2 (1985): 9–15.

Söderström, T. “On Some Algorithms for Design of Optimal Constrained Regulators.” *IEEE Transactions on Automatic Control* AC-23, no. 6 (December 1978): 1100–1101.

Srinathkumar, S. “Eigenvalue/Eigenvector Assignment Using Output Feedback.” *IEEE Transactions on Automatic Control* AC-23, no. 1 (February 1978): 79–81.

Stern, G., and H. A. Henke. “A Design Procedure and Handling-Quality Criteria for Lateral-Directional Flight Control Systems.” *AF-FDL-TR-70-152*. Wright-Patterson AFB, Ohio: Air Force Flight Dynamics Laboratory, May 1971.

Stevens, B. L., F. L. Lewis, and F. Al-Sunni. "Aircraft Flight Controls Design Using Output Feedback." *Journal of Guidance, Control, and Dynamics*, 15, no. 1 (January–February 1992): 238–246.

Tomlin, C., J. Lygeros, L. Benvenuti, and S. Sastry. "Output Tracking for a Non-Minimum Phase Dynamic CTOL Aircraft Model." *Proceedings of the Conference on Decision and Control*, New Orleans, December 1995, pp. 1867–1872.

Wright Laboratory Report WL-TR-96-3099. Wright-Patterson AFB, Ohio, May 1996.

PROBLEMS

Section 5.2

5.2-1 Eigenstructure Assignment with Full State Feedback. The short-period approximation of an aircraft with the cg far aft might be described by

$$\dot{x} = \begin{bmatrix} -1.10188 & 0.90528 & -0.00212 \\ 4.0639 & -0.77013 & -0.169190 \\ 0 & 0 & -10 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 10 & 0 \end{bmatrix} u, \qquad (1)$$

which includes an elevator actuator of $10/(\mathrm{s}+10)$. The state is $x = [\alpha \quad q \quad \delta_e]^{\mathrm{T}}$. An extra control input u_2 has been added to illustrate the extra design freedom available in multivariable systems.

(a) Find the poles.

(b) To conform to flying qualities specifications, it is desired to assign closed-loop short-period eigenvalues λ_1 and λ_2 of $-2 \pm j2$. The actuator pole does not matter but may be assigned to $s = -15$ to speed up its response. The desired closed-loop eigenvectors are

$$v_1 = v_2^* = [0.20 + j0.35 - 0.98 + j0.07 \quad 0]^{\mathrm{T}},$$
$$v_3 = [0 \quad 0 \quad 1]^{\mathrm{T}}$$

Find the state feedback gain K in $u = -Kx$ to assign the desired eigenstructure.

5.2-2 Eigenstructure Assignment with Output Feedback. In Problem 5.2-1, a more realistic situation occurs when only measurements of α and q are taken. Then the control is $u = -Ky$ with $y = [\alpha \quad q]^{\mathrm{T}}$. Only two poles may now be assigned. Select desired closed-loop poles as λ_1 and λ_2 in Problem 5.1, with the same eigenvectors v_1 and v_2. Find the required output feedback gain K. Find the closed-loop poles. What happens to the actuator pole?

5.2-3 In Problem 5.2-1, change the control input to $B = [0 \quad 0 \quad 10]^T$ and use feedback of the output $y = [\alpha \quad q]^T$. Now two poles can be assigned, but there is no freedom in selecting the eigenvectors. Select the desired closed-loop poles $\lambda_1 = \lambda_2^* = -2 + j2$. Find the achievable associated eigenvectors v_1 and v_2. Find the feedback gain K. Find the closed-loop actuator pole.

Section 5.3

5.3-1 Fill in the details in the derivation of the design equations in Table 5.3-1.

5.3-2 Output Feedback Design for Scalar Systems

(a) Consider the case where $x(t)$, $u(t)$, $y(t)$ are all scalars. Show that the solution S to the second Lyapunov equation in Table 5.3-1 is not needed to determine the output feedback gain K. Find an explicit solution for P and hence for the optimal gain K.

(b) Repeat for the case where $x(t)$ and $y(t)$ are scalars but $u(t)$ is an m-vector.

5.3-3 Use (5.3-28) to eliminate K in the Lyapunov equations of Table 5.3-1, hence deriving two coupled nonlinear equations that may be solved for the optimal auxiliary matrices S and P. Does this simplify the solution of the output feedback design problem?

5.3-4 Software for Output Feedback Design. Write a program that finds the gain K minimizing the PI in Table 5.3-1 using the Simplex algorithm of Press et al. (1986). Use it to verify the results of Example 5.3-1. Can you tune the elements of Q and R to obtain better closed-loop responses than the ones given?

5.3-5 For the system

$$\dot{x}\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}x+\begin{bmatrix}0\\ 1\end{bmatrix}u, \quad y=[1 \quad 1]x \tag{1}$$

find the output feedback gain that minimizes the PI in Table 5.3-1 with $Q = I$. Try various values of R to obtain a good response. You will need the software from Problem 5.3-4. The closed-loop step response may be plotted using the software described in Appendix B. (Note that system (1) is nothing but Newton's law, since if $x = [p \quad v]^T$, then $\ddot{p} = u$, where $u(t)$ may be interpreted as an acceleration input F/m.)

5.3-6 Gradient-Based Software for Output Feedback Design. Write a program that finds the gain K minimizing the PI in Table 5.3-1 using the Davidon-Fletcher-Powell algorithm (Press et al., 1986). Use it to verify the results of Example 5.3-1.

Section 5.4

5.4-1 Derive (5.4-31).

5.4-2 Derive the necessary conditions in Table 5.4-1.

5.4-3 In Example 5.4-1, use the observability matrix to verify that the original proposed value of $Q = H^{\mathrm{T}}H$ has $(\sqrt{Q}, A)$ unobservable while the Q that contains a (5,5) element has $(\sqrt{Q}, A)$ observable.

5.4-4 Software for LQ Output Feedback Design. Write a program to solve for the optimal gain K in Table 5.4-1 using the Simplex algorithm (Press et al., 1986). Use it to verify Example 5.4-1.

5.4-5 In Example 5.4-1 we used an output with four components. There is an extra degree of freedom in the choice of control gains that may not be needed. Redo the example using the software from Problem 5.4-4, with the output defined as $y = [\alpha_F \quad q \quad \epsilon]^{\mathrm{T}}$.

5.4-6 To see whether the angle-of-attack filter in Example 5.4-1 complicates the design, redo the example using $y = [\alpha \quad q \quad e \quad \in]^{\mathrm{T}}.$

5.4-7 Redo Example 5.4-1 using root-locus techniques like those in Chapter 4. Based on this, are the gains selected by the LQ algorithm sensible from the point of view of classical control theory?

5.4-8 **Gradient-Based Software for LQ Output Feedback Design.** Write a program to solve for the optimal gain K in Table 5.4-1 using the Davidon-Fletcher-Powell algorithms (Press et al., 1986). Use it to verify Example 5.4-1.

5.4-9 **Gain Scheduling.** To implement a control law on an aircraft, it must be gain scheduled over the flight envelope where it will be used. In Section 3.5 a software longitudinal model was given for a transport aircraft. In Section 3.6 it was shown how to use a trim program to obtain linearized state-variable models at different trim conditions. Using the trim software, obtain three state-variable models for the short-period approximation at 0 ft altitude for speeds of 170, 220, and 300 ft/s. Redo the normal acceleration CAS in Example 5.4-1 for each of these three state-space models. The result is three sets of control gains, each of which is valid for one of the trim conditions. To implement the gain-scheduled control law, write a simple program that selects between the control gains depending on the actual measured speed of the aircraft. Use linear interpolation between the three gain element values for points between the three equilibrium conditions.

Section 5.5

5.5-1 Show the validity of (5.5-8) and (5.5-9).

5.5-2 Use a technique like that employed in Section 5.3 to derive the expression for the optimal cost in terms of P_k that appears in Table 5.5-1. You will need to successively integrate by parts (MacFarlane, 1963).

5.5-3 Derive the necessary conditions in Table 5.5-1.

5.5-4 **Software for Output Feedback LQR Design.** Write a program that finds the gain K minimizing the PI in Table 5.3-1 using the Simplex algorithm (Press et al., 1986). Include gain element weighting using (5.5-6). Use this software to verify the results of Example 5.5-1.

5.5-5 **Software for Output Feedback LQ Tracker Design.** Write a program that finds the gain K minimizing the PI in Table 5.4-1 using the Simplex algorithm (Press et al., 1986). Include gain element weighting using (5.5-6).

5.5-6 In Example 5.4-1 we used an output with four components. There is an extra degree of freedom in the choice of control gains which may not be needed. Using the gain element weighting software from Problem 5.5-5, redo the example with a large weight on the gain element multiplying $e(t)$ to drive it to zero. Is the performance as good? Try tuning the performance index weights for better performance.

5.5-7 **Software for Time-Weighted Output Feedback Tracker Design.** Write a program that finds the gain K minimizing the PI in Table 5.5-1 using the Simplex algorithm (Press et al., 1986). Include gain element weighting using (5.5-6). Use this software to verify the results of Example 5.5-2. Redo the design using weighting of t^3, t^4. Is there any significant difference from the t^2 case?

5.5-8 **Root-Locus Design.** Redo Example 5.5-3, finding the control gains using root-locus techniques like those in Chapter 4. Compare this procedure to modern design using software that solves the design equations in Table 5.5-1.

Section 5.6

5.6-1 Derive the implicit model-following design equations (5.6-43) to (5.6-45).

5.6-2 Using the control gains found in the flare control system of Example 5.6-1, determine the compensator zeros in Figure 5.6-3.

5.6-3 A system obeying Newton's laws is described by the state equations

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad y = [1 \quad 1]x$$

The state is $x = [p \quad v]^{\mathrm{T}}$, with $p(t)$ the position and $v(t)$ the velocity.
Using the CGT approach, design an explicit model-following controller that makes the position follow a quadratic input command $r(t) = r_0 + r_1 t + r_2 t^2$.

5.6-4 It is desired to make the scalar plant

$$\dot{x} = x + u, \quad y = x, \quad z = x$$

behave like the scalar model

$$\dot{\underline{x}} = -2\underline{x} + r, \quad \underline{y} = \underline{x}, \quad \underline{z} = \underline{x}$$

with reference input r equal to the unit step. Use explicit model following to design a servosystem:

(a) Draw the controller structure.

(b) Select the control gains using LQR design on the augmented system.

Section 5.7

5.7-1 Damped Harmonic Oscillator. The damped harmonic oscillator is described by

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u,$$

with ζ the damping ratio and ω_n the natural frequency. This system is useful in modeling systems with an oscillatory mode (e.g., short-period mode, fuel slosh).

(a) Repeat Example 5.7-1 for this system.

(b) For several choices of the PI weighting parameters, find the optimal gain and simulate the closed-loop response. (You can check your results using the software written to solve the design equations in Table 5.3-1, 5.4-1, or 5.5-1 by setting $C = I$ there.)

Section 5.8

5.8-1 A basic helicopter model (Shahian and Hassul, 1993) is given by

$$\dot{x} = \begin{bmatrix} -0.4 & 0 & -0.01 \\ 1 & 0 & 0 \\ -1.4 & 9.8 & -0.02 \end{bmatrix} x + \begin{bmatrix} 6.3 \\ 0 \\ 9.8 \end{bmatrix} \delta,$$

where the state is $x = [q \quad \theta \quad v]$, with $q =$ pitch rate, $\theta =$ pitch angle, and $v =$ horizontal velocity. The control input is the rotor tilt angle δ.

(a) Select different controlled variables as outputs and investigate the stability of the zero dynamics.

(b) Select a *CV* that yields stable zero dynamics. Design the dynamic inversion controller. Simulate using MATLAB.

5.8-2 A nonlinear system is given by

$$\begin{aligned} \dot{x}_1 &= x_1 x_2 + x_3 \\ \dot{x}_2 &= -2x_2 + x_1 u \\ \dot{x}_3 &= \sin x_1 + 2x_1 x_2 + u \end{aligned}$$

(a) Select $y(t) = x_1(t)$ as the controlled variable. Investigate the stability of the zero dynamics.

(b) Design the dynamic inversion controller. Simulate using MATLAB.

5.8-3 Perform the full simulation in Example 5.8-2. That is, combine the dynamic inversion controller and the aircraft into one M file and plot the outputs using MATLAB as done in Example 5.8-1.