

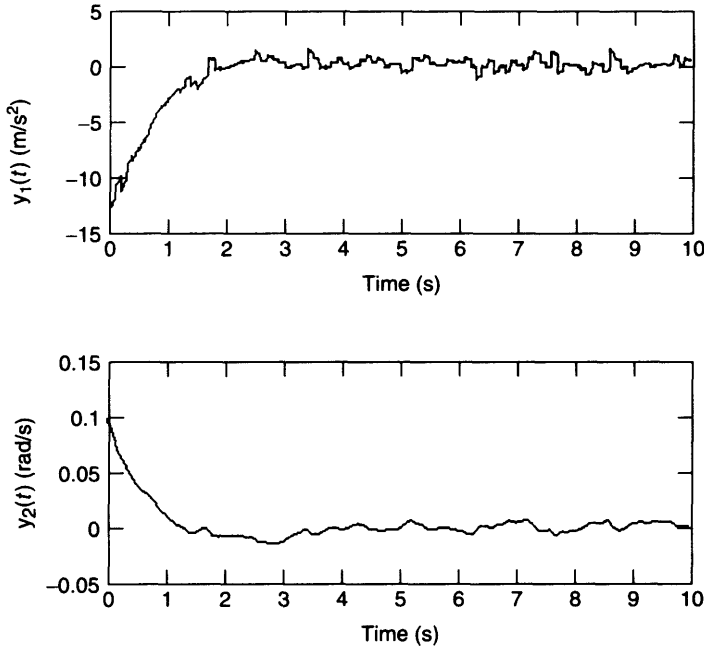


Figure 6.11 Simulated initial response of the flexible bomber with an output-weighted optimal regulator including measurement noise in the pitch channel (SIMULINK block-diagram of Figure 6.3)

6.5 Terminal Time Weighting: Solving the Matrix Riccati Equation

Thus far, we have considered the optimal control problems in which the closed-loop system is brought to a desired state by minimizing the deviation of the *transient response* from the desired state. However, sometimes it is also important to *minimize* the deviation of the system from the desired state at a *specified final time*, t_f . In other words, it is not only important to reach the final state, but to reach there in a *specified final time*, t_f , called the *terminal time*. By assigning a weightage to the state of the system at the terminal time, $\mathbf{x}(t_f)$, in the objective function to be minimized, we can specify the relative importance of minimizing the deviation from the final state. The objective function with terminal time weighting can be expressed as follows:

$$J(t, t_f) = \int_t^{t_f} [\mathbf{x}^T(\tau)\mathbf{Q}(\tau)\mathbf{x}(\tau) + \mathbf{u}^T(\tau)\mathbf{R}(\tau)\mathbf{u}(\tau)] d\tau + \mathbf{x}^T(t_f)\mathbf{V}\mathbf{x}(t_f) \quad (6.77)$$

where $\mathbf{V}$ is the constant *terminal time weighting matrix*. For minimizing the objective function given by Eq. (6.77) for the optimal regulator problem (i.e. when $\mathbf{u}(t) = -\mathbf{K}_o(t)\mathbf{x}(t)$), it can be shown using steps similar to those in Section 6.1, that the optimal feedback gain matrix is given by

$$\mathbf{K}_o(t) = \mathbf{R}^{-1}(t)\mathbf{B}^T(t)\mathbf{M}_o \quad (6.78)$$

where $\mathbf{M}_0$ is the solution to the matrix Riccati equation, Eq. (6.29), repeated here as follows:

$$-\partial \mathbf{M}_0 / \partial t = \mathbf{A}^T(t) \mathbf{M}_0 + \mathbf{M}_0 \mathbf{A}(t) - \mathbf{M}_0 \mathbf{B}(t) \mathbf{R}^{-1}(t) \mathbf{B}^T(t) \mathbf{M}_0 + \mathbf{Q}(t) \quad (6.79)$$

subject to the following *terminal condition*:

$$\mathbf{M}_0(t_f, t_f) = \mathbf{V} \quad (6.80)$$

Hence, the terminal time weighted problem differs from the standard problem of Section 6.1 only in the *non-zero* terminal condition for the matrix Riccati equation.

Note that the terminal time weighted optimal control *cannot* be simplified in a manner similar to the infinite time control, which results in an algebraic Riccati equation. In other words, we cannot avoid taking the bull by the horns anymore; we have to solve the *matrix* Riccati equation if we want to solve the terminal time control problem. For simplicity, let us confine ourselves to *time-invariant* plants only, $\mathbf{A}(t) = \mathbf{A}$, $\mathbf{B}(t) = \mathbf{B}$, and $\mathbf{Q}(t) = \mathbf{Q}$, $\mathbf{R}(t) = \mathbf{R}$, where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, and $\mathbf{R}$ are constants. Then the matrix Riccati equation becomes

$$-\partial \mathbf{M}_0 / \partial t = \mathbf{A}^T \mathbf{M}_0 + \mathbf{M}_0 \mathbf{A} - \mathbf{M}_0 \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{M}_0 + \mathbf{Q} \quad (6.81)$$

which is to be solved with the terminal condition given by Eq. (6.80). We remarked in Section 6.1 on the difficulty of solving the matrix Riccati equation, owing to its *nonlinear* nature. However, if we look carefully at Eq. (6.81), we find that by expressing $\mathbf{M}_0$ in the following manner

$$\mathbf{M}_0(t, t_f) = \mathbf{E}(t, t_f) \mathbf{F}^{-1}(t, t_f) \quad (6.82)$$

where $\mathbf{E}(t, t_f)$ and $\mathbf{F}(t, t_f)$ are two unknown matrices, we can write Eq. (6.81), as the following set of coupled, *linear* differential equations:

$$\begin{bmatrix} \partial \mathbf{F}(t, t_f) / \partial t \\ \partial \mathbf{E}(t, t_f) / \partial t \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \mathbf{F}(t, t_f) \\ \mathbf{E}(t, t_f) \end{bmatrix} \quad (6.83)$$

If we define a matrix $\mathbf{H}$, called the *Hamiltonian*, as follows

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \\ -\mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \quad (6.84)$$

then we can write the solution to the homogeneous set of first order, *matrix* differential equation, Eq. (6.83), as an extension of the solution given by Eq. (4.16) to the homogeneous, *vector* state-equation. Hence, we can write

$$\begin{bmatrix} \mathbf{F}(t, t_f) \\ \mathbf{E}(t, t_f) \end{bmatrix} = \exp\{\mathbf{H}(t - t_f)\} \begin{bmatrix} \mathbf{F}(t_f, t_f) \\ \mathbf{E}(t_f, t_f) \end{bmatrix} \quad (6.85)$$

where $\exp\{\mathbf{H}(t - t_f)\}$ is a *matrix exponential* which can be calculated by the methods of Chapter 4. To satisfy the terminal condition, Eq. (6.80), we can choose $\mathbf{E}(t_f, t_f) = \mathbf{V}$, and $\mathbf{F}(t_f, t_f) = \mathbf{I}$. Thus, for a given time, t , the matrices $\mathbf{E}(t, t_f)$ and $\mathbf{F}(t, t_f)$ can be

calculated using Eq. (6.85), where the matrix exponential, $\exp\{\mathbf{H}(t - t_f)\}$, is calculated by an appropriate algorithm, such as *expm*, *expm1*, *expm2*, or *expm3* of MATLAB. However, care must be taken to select $\mathbf{Q}$ and $\mathbf{R}$ matrices, such that the calculated matrix $\mathbf{F}(t, t_f)$ is *non-singular*, otherwise $\mathbf{M}_0$ cannot be computed using Eq. (6.82). A MATLAB M-file called *matricc.m*, which solves the matrix Riccati equation for linear, time-invariant systems using Eq. (6.85) is given in Table 6.1, and can be called as follows:

```
>>Mo = matricc(A,B,Q,R,V,t,tf,dt) <enter>
```

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{V}$, t , and $tf(t_f)$ are the same as in Eq. (6.77), dt is the time-step, and $\mathbf{M}_0(\mathbf{M}_0)$ is the returned solution of the matrix Riccati equation, Eq. (6.81).

For solving the terminal time weighted optimal control problem, it is clear that while the matrix Riccati equation is solved by marching *backwards* in time using Eq. (6.85) – starting with the *terminal condition*, Eq. (6.82) – the state-equation of the closed-loop system must be solved by marching *forwards* in time using the methods of Chapter 4, starting with the *initial condition*, $\mathbf{x}(t_0) = \mathbf{x}_0$. Hence, the solution requires marching in both forward and backward direction (in time), beginning from the conditions specified at the two time boundaries, i.e. the terminal time, t_f , and initial time, t_0 . For this reason, the general optimal control problem is referred to as *two-point boundary-value problem*. We can write a computer program combining the solution to the matrix Riccati equation and the solution to the closed-loop state-equation. Such a program is given in Table 6.2 as an M-file called *tpbvlti.m*, which solves the two-point boundary-value

Table 6.1 Listing of the M-file *matricc.m* for the solution of the terminal-time weighted, linear optimal control problem

```
matricc.m
```

```
function M=matricc(A,B,Q,R,V,t,tf);
% Program for solving the Matrix Riccati equation resulting
% from the terminal-time weighted optimal control problem for
% linear, time-invariant systems.
% A = State dynamics matrix of the plant
% B = Input coefficient matrix of the plant
% Q = State weighting matrix
% R = Control weighting matrix
% V = Terminal state weighting matrix
% t = present time
% tf = terminal time
% M = returned solution of the Matrix Riccati equation
% Copyright(c)2000 by Ashish Tewari
%
% Construct the Hamiltonian matrix:-
H=[A -B*inv(R)*B';-Q -A'];
% Solve the Matrix Riccati equation using the matrix exponential:-
n=size(A,1);
FE=expm2(H*(t-tf))*[eye(n);V];
F=FE(1:n,:);E=FE(n+1:2*n,:);
M=E*inv(F);
```

Table 6.2 Listing of the M-file *tpbvlti.m* for solving the two-point boundary value problem associated with the solution of the closed-loop state-equations with a terminal-time weighted, linear time-invariant, optimal regulator

tpbvlti.m

```
function [u,X,t] = tpbvlti(A,B,Q,R,V,t0,tf,X0)
% Time-marching solution of the two-point boundary-value problem
% resulting from the terminal-time weighted, optimal regulator
% for a linear, time-invariant system.
% A= State dynamics matrix of the plant
% B= Input coefficient matrix of the plant
% Q= State weighting matrix
% R= Control weighting matrix
% V= Terminal state weighting matrix
% t0= initial time; tf= terminal time (tf-t0 should be small for
% convergence)
% X0= initial state vector; t= time vector
% u=matrix with the ith input stored in the ith column, and jth row
% corresponding to the jth time point
% X= returned matrix with the ith state variable stored in the ith column,
% and jth row corresponding to the jth time point
% copyright(c)2000 by Ashish Tewari
[w,z]=damp(A);
mw=max(w);
if mw==0;
dt=(tf-t0)/20;
else
dt=1/max(w);
end
t=t0:dt:tf;
n=size(t,2);
% initial condition:-
X(1,:)=X0';
% solution of the matrix Riccati equation for t=t0:-
M=matricc(A,B,Q,R,V,t0,tf);
% calculation of input vector for t=t0:-
u(1,:)=-X(1,:)*M*B*inv(R);
% beginning of the time-loop:-
for i=1:n-1
% solution of the matrix Riccati equation:-
M=matricc(A,B,Q,R,V,t(i),tf);
% calculation of the closed-loop state-dynamics matrix
Ac=A-B*inv(R)*B'*M;
% conversion of system from continuous-time to digital
[ad,bd]=c2d(Ac,B,dt);
% solution of digitized closed-loop state-equations
X(i+1,:)=X(i,:)*ad';
% updating the input vector for time t=t(i):-
u(i+1,:)=-X(i+1,:)*M*B*inv(R);
end
```

problem for a linear, *time-invariant* plant using *matricc.m* for the solution of matrix Riccati equation. This M-file can be used as follows:

```
>>[u,X,t] = tpbvlti(A,B,Q,R,V,t0,tf,X0) <enter>
```

where **A**, **B**, **Q**, **R**, **V**, t , and $t_f(t_f)$ have their usual meanings, t_0 is the initial time at which the initial condition vector, $\mathbf{x}_0$, is specified, and **u**, **x**, and **t** contain the returned input, state and time vectors, respectively. However, the usage of this M-file is restricted to small value of the interval $t_f - t_0$. For larger time intervals, a time-marching procedure of Chapter 4 could be used, employing $t_f - t_0$ as the time step.

Example 6.6

Terminal time weighting is very common in problems where two objects are desired to be brought together in a specified final time, such as missile guidance to a target, a rendezvous (or docking) of two spacecraft, or a rendezvous of an aircraft and a tanker aircraft for refuelling. When posing the optimal guidance strategy for such problems, usually the state variables are the relative distance between the two objects, $x_1(t)$, and the relative velocity (also called the *closure velocity*), $x_2(t)$. At some specified final time, t_f , it is desired that the relative distance becomes zero, i.e. $x_1(t_f) = 0$. Whereas in a rendezvous problem, it is also desired that the final closure velocity also becomes zero, i.e. $x_2(t_f) = 0$, such a condition is not imposed on guiding a missile to its target. If $x_1(t_f) \neq 0$, a *miss* is said to occur, and the *miss distance*, $x_1(t_f)$, is a measure of the success (or failure) of either the rendezvous or the missile intercept. The linear, time-invariant state-equation for a missile guidance or a rendezvous problem can be written in terms of the state variables, $x_1(t)$ and $x_2(t)$, and single input, $u(t)$ – which is the *normal acceleration* provided to the object – with the following state coefficient matrices:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (6.86)$$

When we do not care how the missile (or spacecraft) moves before the intercept (or rendezvous) occurs, we may not wish to assign any weightage to the transient response by choosing $\mathbf{Q} = \mathbf{0}$, and $\mathbf{R} = 1$. In such cases, the matrix Riccati equation can be solved *analytically*, and you may refer to Bryson and Ho [2] for the resulting exact solutions. However, it is generally desired that the plant adheres to certain limitations in its transient response, while proceeding from the initial time to the final time, hence $\mathbf{Q}$ is appropriately chosen to be non-zero. Let us consider a spacecraft docking (rendezvous) problem. The initial relative distance and closure velocity between the two spacecraft are 100 m and -10 m/s, respectively. It is desired to complete the rendezvous in 10 seconds, with the closure velocity never exceeding a magnitude of 50 m/s, while requiring a maximum normal acceleration input magnitude of less than 50 m/s^2 . The docking will be considered successful if the final magnitudes of relative distance, $x_1(t_f)$, and relative velocity, $x_2(t_f)$, are less than 0.15 m and 0.5 m/s, respectively. Since both relative distance,

$x_1(t)$, and closure velocity, $x_2(t)$, are required to be minimized at the terminal time, the terminal time weighting matrix, $\mathbf{V}$, should assign weightage to both the state variables. The following choice of $\mathbf{Q}$, $\mathbf{R}$, and $\mathbf{V}$ is seen to meet the performance requirements:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}; \quad \mathbf{R} = 3; \quad \mathbf{V} = \begin{bmatrix} 10 & 0 \\ 0 & 0 \end{bmatrix} \quad (6.87)$$

and the two-point boundary-value problem is solved using *tpbvlti.m* as follows:

```
>>[u,X,t] = tpbvlti(A,B,[1 0; 0 0],3,[10 0; 0 0],0,10,
[100 -10]'); <enter>
```

The calculated state variables, $x_1(t)$ and $x_2(t)$, and the input, $u(t)$, are plotted in Figure 6.12. Note that the performance objectives are met quite successfully. The calculated values of $x_1(t_f)$ and $x_2(t_f)$ are not *exactly* zeros, but -0.1438 m and 0.3610 m/s, respectively; however, these are small enough to be acceptable.

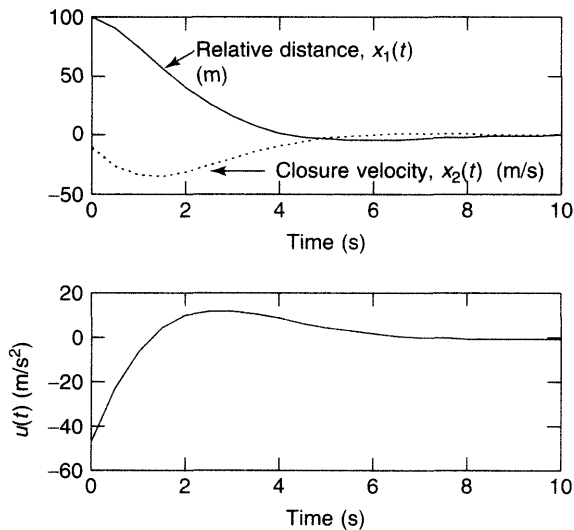


Figure 6.12 Closed-loop docking of spacecraft with terminal time weighted, optimal regulator

Example 6.7

Suppose it is desired to bring both normal acceleration, $y_1(t)$, and the pitch-rate, $y_2(t)$, of the flexible bomber aircraft (Example 6.5) to zero in *exactly* 0.2 seconds, after encountering a perturbation in the pitch-rate of 0.1 rad/s, *regardless* of the maximum overshoot and settling time of the transient response. Since we do not care what happens to the transient response, we can choose $\mathbf{Q} = \mathbf{0}$. Then the choice of