Introduction to Linear Quadratic Optimal Control

- Up to now, we've focused on obtaining the control function, $\boldsymbol{u}$, which optimizes the specified cost as a function of time (along with $\boldsymbol{x}$ and $\boldsymbol{\lambda}$)
  - typically, the problem was solved for a given set of initial conditions in order to proceed to a specified set of terminal constraints
  - so, provided we stay on the optimal path, all we have to do is check our watch and apply the appropriate control

$\Rightarrow$ OPEN-LOOP CONTROL!

- But what do we do if (i) disturbances knock us off the optimal path, or (ii) we don't know exactly where we will start from, or (iii) we don't know exactly when we will start?
  - EITHER, we must solve a new problem for each different situation,
  - OR we can calculate a family of optimal paths so that all possible starting conditions lie on or very close to one of the paths
    - in this second case, we can simply look at where we are and decide what to do next

$\Rightarrow$ FEEDBACK CONTROL

Example: Zermelo's Problem

- Recall the solution:

$$\frac{x}{h} = \frac{1}{2}\left\{\sec\theta_f\left(\tan\theta - \tan\theta_f\right) + \tan\theta\left(\sec\theta_f - \sec\theta\right) + \ln\left[\frac{\tan\theta + \sec\theta}{\tan\theta_f + \sec\theta_f}\right]\right\}$$

$$\frac{y}{h} = \sec\theta - \sec\theta_f$$

  – so, given the current x and y values, then θ and θ_f can be calculated:

$$\tan\theta = \tan\theta_f - \frac{V\left(t_f - t\right)}{h}$$

  – we can thus go backward in time to identify θ, x, and y for any given $\theta_f \Rightarrow$ a family of optimal paths

- Obviously, the problem of generating feedback control solutions using the techniques suggested above will, in general, be an extremely tedious one

  – one saving grace: a unique optimal control vector will, in general, be associated with each point; so we don't have to worry about selecting the proper solution from many alternatives

  – still, the solution process is rather laborious

- Is there another way? YES $\Rightarrow$ DYNAMIC PROGRAMMING

Dynamic Programming

- Basis for the approach: if we start from a given point and proceed optimally to the end, there will be a unique optimal value for the cost associated with this process (J^*)
  - clearly, J^* is a function of the initial point; so J^* is often referred to as the OPTIMAL RETURN FUNCTION
  - using Hamilton-Jacobi theory, the solution of a special partial differential equation that is satisfied by J^* can be used to determine the optimal feedback control policy
  - furthermore, this theory has been generalized to include multistage systems and combinatorial problems by Bellman to produce the complete Dynamic Programming approach
- The complete development of this theory is beyond the scope of this course, but I do want to highlight the result, an important interpretation of the result, and the significance of (and difficulties with) the theory
  - an in-depth treatment of this approach is developed in ECE 5530, Multivariable Control Systems II

RESULT: *Hamilton-Jacobi-Bellman partial differential equation*

- The optimal control policy is given by the solution of

$$-\frac{\partial J^*}{\partial t} = H^*\left(\boldsymbol{x},\ \frac{\partial J^*}{\partial \boldsymbol{x}},\ t\right)$$

where

$$H^*\left(\boldsymbol{x},\ \frac{\partial J^*}{\partial \boldsymbol{x}},\ t\right) = \min_u H\left(\boldsymbol{x},\ \frac{\partial J^*}{\partial \boldsymbol{x}},\ u,\ t\right)$$

- this is an alternative approach to the Calculus of Variations for solving dynamic optimization problems
- in fact, the Euler-Lagrange equations can be developed from the Hamilton-Jacobi-Bellman equation

INTERPRETATION: *The principle of optimality*

> "An optimal policy has the property that, no matter what the previous controls have been, the remaining controls must constitute an optimal policy with regard to the states resulting from the previous control"

SIGNIFICANCE:

1. Emphasizes the existence of optimal feedback control laws
2. Provides a straightforward approach to solving combinatorial problems (e.g., Bryson & Ho, pp. 136-141)

DIFFICULTIES:

- It is not generally feasible to solve the Hamilton-Jacobi-Bellman partial differential equation for practical nonlinear systems
  - so the development of *exact* feedback control schemes is typically out of reach
  - but, if we focus on linear dynamic systems and impose quadratic performance criteria and constraints, appropriate feedback controllers can be synthesized

$\Rightarrow$ LINEAR QUADRATIC OPTIMAL CONTROL

- furthermore, once the LQ techniques have been developed, they may be applied to nonlinear problems via "perturbation guidance" $\Rightarrow$ identify optimal feedback paths in the neighborhood of a previously-identified nominal optimal path

Linear Quadratic Control Problem

- In this section, we examine the Linear Quadratic Control problem and develop techniques to generate optimal feedback control laws
- Dynamic System Description:

$$\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{u}(t)$$
$$\boldsymbol{y}(t) = C(t)\boldsymbol{x}(t) + D(t)\boldsymbol{u}(t)$$

- Types of Control Algorithms and Associated Costs:

1. Terminal Controller $\Rightarrow$ designed to bring a system close to desired conditions at some specified (or unspecified) terminal time

(a) "soft" end constraints

$$J = \frac{1}{2}\boldsymbol{x}^T(t_f)P_f\boldsymbol{x}(t_f) + \frac{1}{2}\int_{t_0}^{t_f}\left\{\boldsymbol{x}^TQ\boldsymbol{x} + \boldsymbol{u}^TR\boldsymbol{u}\right\}dt$$

(b) "hard" end constraints

$$J = \frac{1}{2}\int_{t_0}^{t_f}\left\{\boldsymbol{x}^TQ\boldsymbol{x} + \boldsymbol{u}^TR\boldsymbol{u}\right\}dt$$

$$\boldsymbol{x}_i(t_f) = 0; \qquad i = 1, 2, \ldots, q$$

2. Regulator $\Rightarrow$ designed to keep a stationary system within an acceptable deviation from a reference condition using acceptable amounts of control

$$J = \frac{1}{2}\int_{t_0}^{t_f} \left\{\boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u}\right\} dt \quad \text{with } \left(t_f - t_0\right) \to \infty$$

- We begin our investigation with the simplest form of the linear quadratic optimal control problem $\Rightarrow$ the "soft" end constraint problem presented above:

$$J = \frac{1}{2}\boldsymbol{x}^T(t_f)P_f\boldsymbol{x}(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left\{\boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u}\right\} dt$$

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}$$

$$P_f \geq 0 \qquad Q \geq 0 \qquad R > 0$$

  – the selection of P_f, Q, and R is based on our desire to obtain "acceptable" levels for $\boldsymbol{x}(t_f)$, $\boldsymbol{x}(t)$, and $\boldsymbol{u}(t)$, respectively
    - typically, these matrices will be diagonal
    - some rules of thumb for selecting these matrices are:
      - $\dfrac{1}{P_f(i,i)}$ = max acceptable value of $x_i^2(t_f)$
      - $\dfrac{1}{Q(i,i)} = \left(t_f - t_0\right) *$ max acceptable value of $x_i^2(t)$
      - $\dfrac{1}{R(i,i)} = \left(t_f - t_0\right) *$ max acceptable value of $u_i^2(t)$

- How do we solve this problem? Using the Calculus of Variations

$$\bar{J} = \frac{1}{2}\boldsymbol{x}_f^T P_f \boldsymbol{x}_f + \int_{t_0}^{t_f} \left\{\frac{1}{2}\left[\boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u}\right] + \boldsymbol{\lambda}^T\left[A\boldsymbol{x} + B\boldsymbol{u} - \dot{\boldsymbol{x}}\right]\right\} dt$$

$$\delta\bar{J} = \boldsymbol{x}_f^T P_f \delta\boldsymbol{x}(t_f) + \boldsymbol{\lambda}^T(t_0)\delta\boldsymbol{x}(t_0) - \boldsymbol{\lambda}^T(t_f)\delta\boldsymbol{x}(t_f)$$

$$+ \int_{t_0}^{t_f} \left\{\left[\boldsymbol{x}^T Q + \boldsymbol{\lambda}^T A + \dot{\boldsymbol{\lambda}}^T\right]\delta\boldsymbol{x} + \left[\boldsymbol{u}^T R + \boldsymbol{\lambda}^T B\right]\delta\boldsymbol{u}\right\} dt$$

– yielding the following equations:

$$\begin{aligned}\dot{\boldsymbol{\lambda}}^T &= -\boldsymbol{x}^T Q - \boldsymbol{\lambda}^T A \\ \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \boldsymbol{u}^T R + \boldsymbol{\lambda}^T B &= 0 \\ \boldsymbol{\lambda}^T(t_f) &= \boldsymbol{x}_f^T P_f\end{aligned}$$

– rearranging, we have

$$\begin{aligned}\dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \dot{\boldsymbol{\lambda}} &= -Q\boldsymbol{x} - A^T\boldsymbol{\lambda} \\ \boldsymbol{u} &= -R^{-1}B^T\boldsymbol{\lambda}\end{aligned}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \qquad \boldsymbol{\lambda}(t_f) = P_f\boldsymbol{x}_f$$

– NOTE: P_f, Q, and R are symmetric matrices

– we would have gotten the same result using the following equations:

$$\frac{\partial H}{\partial \boldsymbol{u}} = 0 \qquad \frac{\partial H}{\partial \boldsymbol{x}} = -\dot{\boldsymbol{\lambda}}^T \qquad \frac{\partial H}{\partial \boldsymbol{\lambda}} = \dot{\boldsymbol{x}}$$

$$\boldsymbol{\lambda}^T(t_f) = \frac{\partial \varphi}{\partial \boldsymbol{x}(t_f)}$$

where

$$H = \frac{1}{2}\left\{\boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u}\right\} + \boldsymbol{\lambda}^T\left\{A\boldsymbol{x} + B\boldsymbol{u}\right\}$$

$$\varphi = \frac{1}{2}\boldsymbol{x}_f^T P_f \boldsymbol{x}_f$$

– we can solve these equations by augmenting the state vector:

$$\begin{bmatrix}\dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\lambda}}\end{bmatrix} = \left[\begin{array}{c|c} A & -BR^{-1}B^T \\ \hline -Q & -A^T\end{array}\right]\begin{bmatrix}\boldsymbol{x} \\ \boldsymbol{\lambda}\end{bmatrix}$$

- NOTE: this matrix is referred to as the HAMILTONIAN matrix, $\mathcal{H}$
  - so, we must find the solution to this set of $2n$ linear homogeneous differential equations with $\boldsymbol{x}(t_o)$ given and $\boldsymbol{\lambda}(t_f) = P_f \boldsymbol{x}_f$

$$\begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{\lambda}(t) \end{bmatrix} = e^{\mathcal{H}(t-t_0)} \begin{bmatrix} \boldsymbol{x}(t_0) \\ \boldsymbol{\lambda}(t_0) \end{bmatrix} = \Phi(t-t_0) \begin{bmatrix} \boldsymbol{x}(t_0) \\ \boldsymbol{\lambda}(t_0) \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{\lambda}(t) \end{bmatrix} = \begin{bmatrix} \phi_{11}(t-t_0) & \phi_{12}(t-t_0) \\ \phi_{21}(t-t_0) & \phi_{22}(t-t_0) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t_0) \\ \boldsymbol{\lambda}(t_0) \end{bmatrix}$$

$\Rightarrow$

$$\begin{aligned} \boldsymbol{x}(t) &= \phi_{11}\boldsymbol{x}(t_0) + \phi_{12}\boldsymbol{\lambda}(t_0) \\ \boldsymbol{\lambda}(t) &= \phi_{21}\boldsymbol{x}(t_0) + \phi_{22}\boldsymbol{\lambda}(t_0) \end{aligned}$$

- $\boldsymbol{x}(t_0)$ and $\boldsymbol{\lambda}(t_f)$ are known, but $\boldsymbol{\lambda}(t_0)$ and $\boldsymbol{x}(t_f)$ are not

$$\begin{aligned} \boldsymbol{x}(t_f) &= \phi_{11}\left(t_f - t_0\right)\boldsymbol{x}(t_0) + \phi_{12}\left(t_f - t_0\right)\boldsymbol{\lambda}(t_0) \\ \boldsymbol{\lambda}(t_f) &= \phi_{21}\left(t_f - t_0\right)\boldsymbol{x}(t_0) + \phi_{22}\left(t_f - t_0\right)\boldsymbol{\lambda}(t_0) = P_f \boldsymbol{x}(t_f) \end{aligned}$$

$\Rightarrow$

$$\left(P_f\phi_{11} - \phi_{21}\right)\boldsymbol{x}(t_0) = \left(\phi_{22} - P_f\phi_{12}\right)\boldsymbol{\lambda}(t_0)$$

$$\boldsymbol{\lambda}(t_0) = \left(\phi_{22} - P_f\phi_{12}\right)^{-1}\left(P_f\phi_{11} - \phi_{21}\right)\boldsymbol{x}(t_0)$$

- and now that we know $\boldsymbol{x}(t_0)$ and $\boldsymbol{\lambda}(t_0)$, we can calculate $\boldsymbol{x}(t)$, $\boldsymbol{\lambda}(t)$ and $\boldsymbol{u}(t)$

Example:

$$\dot{x} = u$$

$$J = \frac{1}{2}px^2(t_f) + \frac{1}{2}\int_0^{t_f}\left(qx^2 + ru^2\right)dt$$

SOLUTION:

$$H = \frac{1}{2}\left(qx^2 + ru^2\right) + \lambda u$$

$$\frac{\partial H}{\partial u} = ru + \lambda = 0 \quad \Rightarrow \quad u = -\frac{\lambda}{r}$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = -qx \qquad \dot{x} = -\frac{\lambda}{r}$$

$\Rightarrow$

$$\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{r} \\ -q & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix}$$

$$(sI - A)^{-1} = \left(\frac{1}{s^2 - {}^1\!/_{qr}}\right) \begin{bmatrix} s & -\frac{1}{r} \\ -q & s \end{bmatrix}$$

$\Rightarrow$

$$e^{\mathcal{H}t} = \left[\begin{array}{c|c} \cosh\sqrt{q/r}\,t & -1/\sqrt{rq}\,\sinh\sqrt{q/r}\,t \\ \hline -\sqrt{rq}\,\sinh\sqrt{q/r}\,t & \cosh\sqrt{q/r}\,t \end{array}\right]$$

$\Rightarrow$

$$x(t) = x_0 \cosh\sqrt{q/r}\,t - \lambda_0/\sqrt{rq}\,\sinh\sqrt{q/r}\,t$$

$$u(t) = x_0\sqrt{q/r}\,\sinh\sqrt{q/r}\,t - \lambda_0/r\,\cosh\sqrt{q/r}\,t$$

- But how do we find λ_0 ?

$$x_f = x_0 \cosh\sqrt{q/r}\,t_f - \lambda_0/\sqrt{rq}\,\sinh\sqrt{q/r}\,t_f$$

$$\lambda_f = -x_0\sqrt{rq}\,\sinh\sqrt{q/r}\,t_f + \lambda_0 \cosh\sqrt{q/r}\,t_f$$

$\Rightarrow$

$$x_0\left\{p\cosh\sqrt{q/r}\,t_f + \sqrt{rq}\,\sinh\sqrt{q/r}\,t_f\right\}$$

$$= \lambda_0 \left\{ \cosh \sqrt{q/r}\, t_f + c/\sqrt{rq}\, \sinh \sqrt{q/r}\, t_f \right\}$$

$$\lambda_0 = \left\{ \frac{p + \sqrt{rq}\, \tanh \sqrt{q/r}\, t_f}{1 + p/\sqrt{rq}\, \tanh \sqrt{q/r}\, t_f} \right\} x_0$$

- substituting, we find that $u(t)$ is a function of x_0, q, r, p, t_f, and t .
- NOTE: If $q = 0$ and $r = 1$, then

$$\lambda_0 = \left\{ \frac{1}{1 + pt_f} \right\} x_0 = \lambda(t) \quad \Rightarrow \quad u(t) = -\left\{ \frac{p}{r\left(1 + pt_f\right)} \right\} x_0$$

giving constant control input

  - additionally,

$$x_f = \frac{x_0}{1 + pt_f}$$

  so that as $p \to \infty,\ x_f \to 0$

- One problem with this Calculus of Variations approach to solving this problem is that $\boldsymbol{u}(t)$ is a function of time and initial conditions, i.e.

$$\boldsymbol{u}(t) = -R^{-1} B^T \boldsymbol{\lambda}(t)$$

$$\begin{aligned} \boldsymbol{\lambda}(t) &= \phi_{21}(t - t_0)\, \boldsymbol{x}(t_0) + \phi_{22}(t - t_0)\, \boldsymbol{\lambda}(t_0) \\ \boldsymbol{x}(t) &= \phi_{11}(t - t_0)\, \boldsymbol{x}(t_0) + \phi_{12}(t - t_0)\, \boldsymbol{\lambda}(t_0) \end{aligned}$$

$$e^{\mathcal{H}(t-t_0)} = \left[\begin{array}{c|c} \phi_{11} & \phi_{12} \\ \hline \phi_{21} & \phi_{22} \end{array} \right]$$

$$\lambda(t_0) = -M(t_f)\boldsymbol{x}(t_0)$$

$$M(t_f) = \left\{ P_f \phi_{12} - \phi_{22} \right\}^{-1} \left\{ P_f \phi_{11} - \phi_{21} \right\}$$

- It would be nice, however, to identify $\boldsymbol{u}(t)$ in terms of $\boldsymbol{x}(t) \;\Rightarrow\;$ this would give us a FEEDBACK LAW
- Can this be done? YES!

$$\boldsymbol{u}(t) = -R^{-1}B^T\left\{\phi_{21}(t-t_0) - \phi_{22}(t-t_0)\,M(t_f)\right\}\boldsymbol{x}(t_0)$$
$$\boldsymbol{x}(t) = \left\{\phi_{11}(t-t_0) - \phi_{12}(t-t_0)\,M(t_f)\right\}\boldsymbol{x}(t_0)$$

$\Rightarrow$

$$\begin{aligned}\boldsymbol{u}(t) &= -R^{-1}B^T\{\phi_{21} - \phi_{22}M\}\{\phi_{11} - \phi_{12}M\}^{-1}\boldsymbol{x}(t)\\ &= -R^{-1}B^T P(t)\boldsymbol{x}(t) \quad \{\boldsymbol{\lambda}(t) = P(t)\boldsymbol{x}(t)\}\end{aligned}$$

  - NOTE: this approach is not necessarily the best way to solve for $P(t)$, but it does demonstrate the construction of $P(t)$

- Is this result surprising? NO!
  - since $\boldsymbol{u}(t)$ is a function of $\boldsymbol{x}(t_0)$,the Principle of Optimality indicates that any time t may be regarded as a new initial time, t_0
  - so, our optimal control will always be a function of the current states

<u>Example: (cited previously)</u>

$$u(t) = -\frac{c}{1+ct_f}x_0 \quad \Rightarrow \quad u(t) = \left\{-\frac{c}{1+c\left(t_f - t_0\right)}\right\}x(t)$$

- RESULT: The LQ optimal control problem with "soft" end constraints yields a full-state feedback control law:

$$\boldsymbol{u}(t) = -K(t)\boldsymbol{x}(t)$$

  with time-varying control law:

$$K(t) = R^{-1}B^T P(t)$$

- Question: How do we compute $P(t)$?

1. Find $e^{\mathcal{H}t}$ and compute $P(t)$ using the following relationships:

$$\left.\begin{aligned} \boldsymbol{x}(t) &= \phi_{11}\left(t-t_f\right)\boldsymbol{x}(t_f) + \phi_{12}\left(t-t_f\right)\boldsymbol{\lambda}\left(t_f\right) \\ \boldsymbol{\lambda}(t) &= \phi_{21}\left(t-t_f\right)\boldsymbol{x}(t_f) + \phi_{22}\left(t-t_f\right)\boldsymbol{\lambda}\left(t_f\right) \end{aligned}\right\} \; e^{\mathcal{H}(t-t_f)} = \left[\begin{array}{c|c} \phi_{11} & \phi_{12} \\ \hline \phi_{21} & \phi_{22} \end{array}\right]$$

$$\boldsymbol{\lambda}\left(t_f\right) = P_f \boldsymbol{x}(t_f)$$

$$\boldsymbol{\lambda}(t) = \left(\phi_{21} + \phi_{22}P_f\right)\boldsymbol{x}(t_f) \qquad \boldsymbol{x}(t_f) = \left(\phi_{11} + \phi_{12}P_f\right)^{-1}\boldsymbol{x}(t)$$

$$\begin{aligned} \boldsymbol{\lambda}(t) = &\left\{\phi_{21}\left(t-t_f\right) + \phi_{22}\left(t-t_f\right)P_f\right\} \\ &\times \left\{\phi_{11}\left(t-t_f\right) + \phi_{12}\left(t-t_f\right)P_f\right\}^{-1}\boldsymbol{x}(t) \end{aligned}$$

   - Problems:
     - $e^{\mathcal{H}t}$ may be difficult to derive analytically
     - numerical methods may be inaccurate due to different exponetial growth rates within $e^{\mathcal{H}t}$

2. Integrate a “matrix Ricatti equation” (the “sweep” method)

$$\boldsymbol{\lambda}(t) = P(t)\boldsymbol{x}(t) \;\Rightarrow$$

$$\begin{aligned} \dot{\boldsymbol{\lambda}} &= \dot{P}\boldsymbol{x} + P\dot{\boldsymbol{x}} = -A^T\boldsymbol{\lambda} - Q\boldsymbol{x} \\ \dot{\boldsymbol{x}} &= A\boldsymbol{x} - BR^{-1}B^T\boldsymbol{\lambda} \end{aligned}$$

$\Rightarrow$

$$\left\{\dot{P} + PA - PBR^{-1}B^TP\right\}\boldsymbol{x} = \left\{-A^TP - Q\right\}\boldsymbol{x}$$

$$\dot{P} = -PA - A^TP + PBR^{-1}B^TP - Q \quad P(t_f) = P_f$$

NOTE: P will always be a symmetric matrix

– using a numerical integration method, we can let

$$\tau = t_f - t \qquad d\tau = -dt$$

$\Rightarrow$

$$dP = PA + A^T P - PBR^{-1}B^T P + Q$$

and let τ go from 0 to t_f $\left\{P(\tau = 0) = P_f\right\}$

Examples:

1. First-order problem done previously with $q = 0$ and $r = 1$

$$\begin{matrix} A = 0 \\ \\ B = 1 \end{matrix} \quad \Rightarrow \quad \dot{P} = P^2 \quad \Rightarrow \quad \begin{matrix} \dfrac{dP}{P^2} = dt \\ \\ -P^{-1}\Big|_{t_f}^{t} = t - t_f \end{matrix}$$

$$\frac{1}{P_f} - \frac{1}{P(t)} = t - t_f$$

$$\begin{aligned} P(t) &= \frac{1}{1/p + \left(t_f - t\right)} \\ &= \frac{p}{1 + p\left(t_f - t\right)} \end{aligned}$$

$$\mathcal{H} = \begin{bmatrix} 0 & -1/r \\ 0 & 0 \end{bmatrix} \Rightarrow e^{\mathcal{H}t} = \begin{bmatrix} 1 & -t/r \\ 0 & 1 \end{bmatrix}$$

LQ Control: The Regulator Problem

- Remember from the last section that the standard form for the linear quadratic optimal control problem is:

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}$$

$$J = \frac{1}{2}\boldsymbol{x}^T(t_f)P_f\boldsymbol{x}(t_f) + \int_{t_0}^{t_f} \frac{1}{2}\left\{\boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u}\right\} dt$$

  with

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

- Using the Calculus of Variations, we've shown that the optimal control can be written in state feedback form as:

$$\boldsymbol{u}(t) = -K(t)\boldsymbol{x}(t) = -R^{-1}B^T P(t)\boldsymbol{x}(t)$$

  – $\boldsymbol{u}(t)$ is time-varying (depending on time-to-go)
  – $P(t)$ can be identified either by appropriate manipulation of transition matrices or by integrating a matrix Ricatti equation

- Now, we want to focus on a special subset of this category of problems: the regulator problem
  – what is a regulator? A feedback controller designed to keep a stationary system within an acceptable deviation from a reference condition using acceptable amounts of control
    ◦ example: a satellite pointing problem
  – assumptions associated with the regulator problem:
    1. the system is time-invariant (e.g., A and B are constant)
    2. the Q and R matrices in J are constant
    3. $t_f - t_0 \to \infty$

– what are the implications of these assumptions?

- the cost function reduces to

$$J = \frac{1}{2}\int_{t_0}^{t_f} \left(\boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u}\right) dt$$

- the term $1/2\boldsymbol{x}^T(t_f)P_f\boldsymbol{x}(t_f)$ in J can be eliminated since the terminal time is so far into the future; but there must be a running cost on at least some of the states (i.e., $Q \neq 0$) for this problem to be feasible

– under these conditions, a constant, finite solution to the matrix Ricatti equation exists

$$\dot{P} = -PA - A^T P + PBR^{-1}B^T P - Q = 0$$

$$\Rightarrow \; P(t) \to P_0 \text{ as } t_f - t_0 \to \infty$$

- NOTE: this result assumes:
  1. $[A,\ B]$ is stabilizable
  2. $[A,\ C]$ is detectable
  3. $Q = \rho C^T C$
- so, the feedback gain matrix, $K(t)$, will be constant and it can be shown that the controlled system will be stable if $P_0 > 0$

– how do we find P_0 ?

- if $\dot{P} = 0$, the matrix Ricatti equation becomes an algebraic equation that can be solved for the constant matrix P
- but since the resulting equation is quadratic in P, more than one solution will appear $\Rightarrow$ the extraneous solutions can usually be eliminated using the fact that P_0 is positive definite

Example: Bryson & Ho, pg. 168

$$\dot{x} = -\frac{1}{\tau}x + u \qquad J = \frac{1}{2}\int_0^\infty \left(qx^2 + ru^2\right) dt$$

SOLUTION:

$$u(t) = -R^{-1}B^T P_0 x(t) = \frac{-P_0}{r}x(t)$$

$$P = -PA - A^T P + PBR^{-1}B^T P - Q = 0$$

$$\frac{2P_0}{\tau} + \frac{P_0^2}{r} - q = 0$$

$$P_0^2 + \frac{2r}{\tau}P_0 - qr = 0$$

$$P_0 = -\frac{r}{\tau} \pm \sqrt{\left(\frac{r}{\tau}\right)^2 + qr}$$

But $P_0 > 0 \;\Rightarrow$

$$P_0 = \frac{r}{\tau}\left\{\sqrt{1 + q\frac{\tau^2}{r}} - 1\right\}$$

$$u(t) = -\left\{\sqrt{\frac{1}{\tau^2} - \frac{q}{r}} - \frac{1}{\tau}\right\} x(t)$$

- Another method of finding P_0 is to integrate the matrix Riccati equation backwards in time using numerical techniques until P settles down to a constant
  - although $P_f = 0$, the fact that $Q \neq 0$ guarantees the existence of a nonzero P_0
  - unfortunately, this approach is computationally expensive
- Another alternative can be found using calculus of variations approach on a slightly different cost function:

$$J = \frac{1}{2}\int_0^\infty \left\{ \boldsymbol{x}^T C^T Q_1 C \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \right\} dt$$

– why do this? Because I may only be interested in controlling a subset of the states (e.g., the system outputs)

– now, using the calculus of variations:

$$H = \frac{1}{2}\left\{ \boldsymbol{x}^T C^T Q_1 C \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \right\} + \boldsymbol{\lambda}^T (A\boldsymbol{x} + B\boldsymbol{u})$$

$$\dot{\boldsymbol{\lambda}}^T = -\frac{\partial H}{\partial \boldsymbol{x}} \Rightarrow \begin{array}{rl} \dot{\boldsymbol{\lambda}} = & -A^T\boldsymbol{\lambda} - C^T Q_1 C \boldsymbol{x} \\ \dot{\boldsymbol{x}} = & A\boldsymbol{x} + B\boldsymbol{u} \end{array}$$

$$\frac{\partial H}{\partial \boldsymbol{u}} = 0 \Rightarrow \boldsymbol{u} = -R^{-1}B^T\boldsymbol{\lambda} \Rightarrow \dot{\boldsymbol{x}} = A\boldsymbol{x} - BR^{-1}B^T\boldsymbol{\lambda}$$

– so, the optimal solution can be obtained by solving the following set of homogeneous, first-order, linear differential equations:

$$\dot{\boldsymbol{z}} = \mathcal{H}\boldsymbol{z}$$

where

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix} \qquad \mathcal{H} = \left[\begin{array}{c|c} A & -BR^{-1}B^T \\ \hline -C^T Q_1 C & -A^T \end{array}\right]$$

– this result is particularly useful because it can be shown that the $2n$ eigenvalues of $\mathcal{H}$ are symmetric about both the imaginary axis *and* the real axis

- ∘ so, this adjoined system has n stable roots and n unstable roots
- ∘ n roots are associated with $\boldsymbol{x}$ and n roots are associated with $\boldsymbol{\lambda}$
- ∘ which are which? For J to remain finite as $t \to \infty$, $\boldsymbol{x}$ must approach zero

$\Rightarrow$ The n stable eigenvalues of $\mathcal{H}$ must be the closed-loop poles of the system

- NOTE: when we optimize a controllable linear system using a quadratic cost, we will *always* generate a stable, closed-loop system

- Having developed this result, we can now use it to generate the optimal full-state feedback gains

Single-Input-Single-Output Systems: Symmetric Root Locus

- Despite the fact that this is a SISO system, we are assuming that all states are available for feedback

$\Rightarrow$ We have n control gains to select

- If we know the optimal pole locations (by looking at the eigenvalues of $\mathcal{H}$), we can calculate the closed-loop characteristic equation and equate coefficients to identify the optimal control gains

- For SISO systems, there is an easier way to find the optimal pole locations than finding the eigenvalues of $\mathcal{H}$

$$\left.\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \dot{\boldsymbol{\lambda}} &= -A^T\boldsymbol{\lambda} - C^TQ_1C\boldsymbol{x} \end{aligned}\right\} \Rightarrow \begin{aligned} \boldsymbol{x} &= (sI - A)^{-1}B\boldsymbol{u} \\ \boldsymbol{\lambda} &= -(sI - A)^{-1}C^TQ_1C\boldsymbol{x} \end{aligned}$$

$$R\boldsymbol{u} + B^T\boldsymbol{\lambda} = 0$$

$$R\boldsymbol{u} + G^T\left(-sI - A^T\right)^{-1}C^TQ_1C\,(sI - A)^{-1}G\boldsymbol{u} = 0$$

  - since $\boldsymbol{y} = C\boldsymbol{x}$ (assuming here $D = 0$), $Y(s) = C\,(sI - A)^{-1}B \cdot \boldsymbol{U}(s)$

- and from matrix transpose properties,
  $\left(-sI - A^T\right)^{-1} = (-sI - A)^{-T}$
- so, if we define $G(s) = C\,(sI - A)^{-1}\,B$ we get
  $$\left[R + G^T(-s)Q_1G(s)\right]\boldsymbol{U}(s) = 0$$

- For scalar, non-zero $u(t)$, $R + G^T(-s)Q_1G(s)$ is a scalar $2n^{th}$ -order polynomial that is symmetric in s and $-s$ and must equal zero
  - this polynomial is, in fact, the characteristic equation for the Euler-Lagrange equations developed for this problem
  - so, this polynomial can be used to identify the optimal closed-loop poles of the system

- For SISO systems, R and Q_1 are scalars
  $$R + G^T(-s)Q_1G(s) = 0 \;\Rightarrow\; 1 + G^T(-s)\frac{Q_1}{R}G(s) = 0$$
  which is in Root Locus Form (with Q_1/R as the variable gain)
  - thus, root locus techniques can be used to find the optimal closed-loop poles for given ratio Q_1/R
  - the optimal steady-state control gains can then be found by equating coefficients in the closed-loop characteristic equation
  - for SISO systems, this process should be much easier than finding the stable eigenvalues of $\mathcal{H}$

Example:

$$y = -by + u \quad J = \frac{1}{2}\int_0^\infty \left(qy^2 + ru^2\right)dt$$

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s+b}$$

$$\Rightarrow \qquad 1 + \frac{1}{-s+b} \cdot \frac{q}{r} \cdot \frac{1}{s+b} = 0$$

$$\frac{q}{r}\left\{\frac{1}{(s-b)(s+b)}\right\} = 1$$

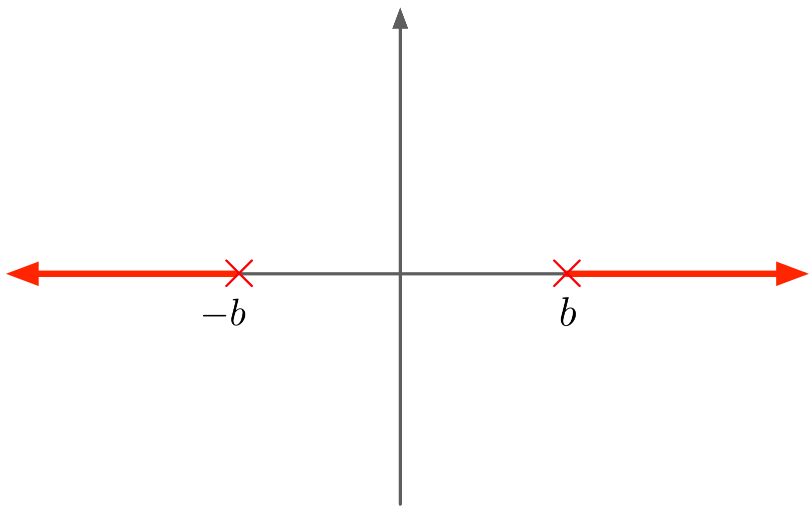

Figure 6.1 0° Root Locus Plot

- Where is the closed-loop pole?

$$s^2 - b^2 - \frac{q}{r} = 0$$

$$s = -\sqrt{b^2 + \frac{q}{r}}$$

- What is the gain?

$$u = -ky \;\; \Rightarrow \;\; s + b + k = 0$$

$$-b - k = -\sqrt{b^2 + \frac{q}{r}}$$

$$k = \sqrt{b^2 + \frac{q}{r}} - b$$

- COMMENT: if $q = 0$ (i.e., no weight on the states), J will only be a function of the control $\Rightarrow$ we are trying to minimize the control
  - $b > 0 \Rightarrow$ open-loop system is stable $\Rightarrow$ no need to use control *and* $k = 0$
  - $b < 0 \Rightarrow$ open-loop system is unstable $\Rightarrow$ finite control is required to stabliize the system *and* $k = 2|b|$
- COMMENT: what if we increase q ?
  - we are trying to tighten our control on the system
  - so the system should become more stable (exactly what the root locus demonstrates)
- Another symmetric root locus example:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \omega^2 \end{bmatrix} u$$

$$y = x_1$$

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\omega^2}{s^2 + \omega^2}$$

$$\Rightarrow \quad 1 + \frac{\omega^2}{(-s)^2 + \omega^2} \cdot \frac{q}{r} \cdot \frac{\omega^2}{s^2 + \omega^2} = 0$$

$$\frac{q}{r} \cdot \frac{\omega^4}{(s^2 + \omega^2)^2} = -1$$

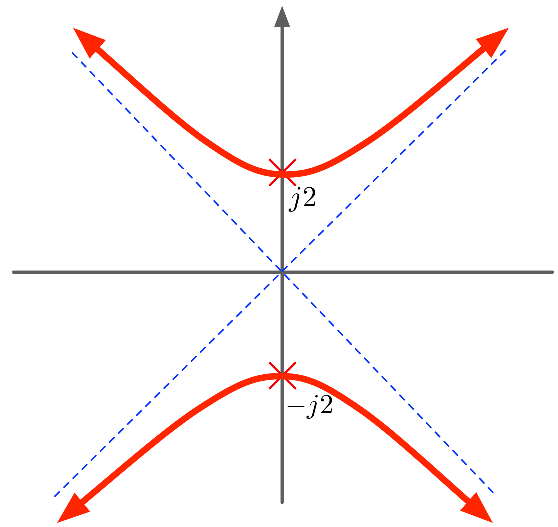

Figure 6.2 180° Root Locus Plot

CHARACTERISTIC EQUATIONS:

$$\left(s^2+\omega^2\right)^2+\frac{q}{r}\omega^4=0 \;\Rightarrow\; s^4+2\omega^2 s^2+\omega^4\left(1+\frac{q}{r}\right)=0$$

$$u=-k_1x_1-k_2x_2 \;\Rightarrow\; \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ -\omega^2(1+k_1) & -\omega^2k_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

$$|sI-A|\,|-sI-A|=s^4+\omega^2\left(2+2k_1-\omega^2k_2\right)s^2+\omega^4(1+k_1)^2=0$$

Multiple-Input-Multiple-Output Systems: Eigenvector Analysis

- For MIMO systems, MacFarlane and Potter developed an elegant eigenvector approach to identify the optimal steady-state feedback control gains
- Since $\boldsymbol{u}=-R^{-1}B^T\boldsymbol{\lambda}$,

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\boldsymbol{\lambda}}\end{bmatrix}=\left[\begin{array}{c|c}A & -BR^{-1}B^T\\ \hline -C^TQ_1C & -A^T\end{array}\right]\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{\lambda}\end{bmatrix}$$

  – and using the eigenvectors of $\mathcal{H}$,

$$\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} = W \left[\begin{array}{c|c} \mathcal{S}_+ & 0 \\ \hline 0 & \mathcal{S}_- \end{array} \right] W^{-1} \begin{bmatrix} x \\ \lambda \end{bmatrix}$$

where $\mathcal{S}_+ = \operatorname{diag}\{+s_i\}$ and $\mathcal{S}_- = \operatorname{diag}\{-s_i\}$, and the columns of W are the eigenvectors of $\mathcal{H}$

– now, let's define a new set of states:

$$z = \begin{bmatrix} z_+ \\ z_- \end{bmatrix} = W_{-1} \begin{bmatrix} x \\ \lambda \end{bmatrix}$$

$$\Rightarrow \quad \dot{z}_+ = \mathcal{S}_+ z_+ \qquad \dot{z}_- = \mathcal{S}_- z_-$$

$$\Rightarrow \quad \begin{bmatrix} z_+(t) \\ z_-(t) \end{bmatrix} = \left[\begin{array}{c|c} e^{-\mathcal{S}_+(t_f - t)} & 0 \\ \hline 0 & e^{-\mathcal{S}_-(t_f - t)} \end{array} \right] \begin{bmatrix} z_+(t_f) \\ z_-(t_f) \end{bmatrix}$$

– based on this solution, we can now solve for $x(t)$ and $\lambda(t)$

$$\begin{bmatrix} x \\ \lambda \end{bmatrix} = T z(t) = \left[\begin{array}{c|c} \mathcal{X}_+ & \mathcal{X}_- \\ \hline \Lambda_+ & \Lambda_- \end{array} \right] \left[\begin{array}{c|c} e^{-\mathcal{S}_+(t_f - t)} & 0 \\ \hline 0 & e^{-\mathcal{S}_-(t_f - t)} \end{array} \right] \begin{bmatrix} z_+(t_f) \\ z_-(t_f) \end{bmatrix}$$

◦ here, $[\mathcal{X}_+,\ \Lambda_+]$ and $[\mathcal{X}_-,\ \Lambda_-]$ represent the eigenvectors associated with $\mathcal{S}_+$and $\mathcal{S}_-$, respectively

$$x(t) = \mathcal{X}_+ e^{-\mathcal{S}_+(t_f - t)} z_+(t_f) + \mathcal{X}_- e^{-\mathcal{S}_-(t_f - t)} z_-(t_f)$$

$$\lambda(t) = \Lambda_+ e^{-\mathcal{S}_+(t_f - t)} z_+(t_f) + \Lambda_- e^{-\mathcal{S}_-(t_f - t)} z_-(t_f)$$

◦ furthermore, as $t_f - t \to \infty$, $\quad e^{-\mathcal{S}_+(t_f - t)} \to 0$

$$x(t) = \mathcal{X}_- e^{-\mathcal{S}_-(t_f - t)} z_-(t_f)$$

$$\lambda(t) = \Lambda_- e^{-\mathcal{S}_-(t_f - t)} z_-(t_f)$$

$$\Rightarrow \quad \lambda(t) = \Lambda_- \mathcal{X}_-^{-1} x(t)$$

$$S_0 = \Lambda_- \mathcal{X}_-^{-1} \;\Rightarrow\; \boldsymbol{u}(t) = -R^{-1}B^T \Lambda_- \mathcal{X}_-^{-1} \boldsymbol{x}(t)$$

- NOTE: For MIMO systems, this algorithm is significantly less computationally expensive than integrating the matrix Ricatti equation

- So far, we've discussed the application of optimal feedback control to systems with "soft" terminal constraints to perform the task of regulation (i.e., keeping the states "close" to zero)
- The next section highlights two alternative control problems:

1. Zero Terminal Error
2. Tracking

Zero Terminal Error Controller

- Consider the following linear quadratic control problem:

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}, \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0$$

$$J = \frac{1}{2}\int_{t_0}^{t_f} \left\{ \boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \right\} dt$$

$$x_i(t_f) = c_i, \qquad (i = 1, 2, \ldots, q)$$

- NOTE: A soft constraint on the other $n - q$ states at t_f could be used, and the following developments could be modified to handle this added complexity. But for now, we'll assume $\varphi = 0$.

- Standard Calculus of Variations approach:

$$\Phi = \sum_{i=1}^{q} \nu_i \left\{ x_i(t_f) - c_i \right\} \qquad H = \frac{1}{2}\left\{ \boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \right\} + \boldsymbol{\lambda}^T \left\{ A\boldsymbol{x} + B\boldsymbol{u} \right\}$$

$$\dot{\boldsymbol{\lambda}} = -Q\boldsymbol{x} - A^T\boldsymbol{\lambda}, \qquad \lambda_i(t_f) = \begin{cases} \nu_i, & i = 1, 2, \ldots, q \\ 0, & i = q+1, \ldots, n \end{cases}$$

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}, \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0$$

$$x_i(t_f) = c_i, \quad i = 1, 2, \ldots, q$$

$$\boldsymbol{u} = -R^{-1}B^T\boldsymbol{\lambda}$$

– the solution is obtained by solving the following equations:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix}$$

$$x_i(t_f) = c_i, \quad i = 1, 2, \ldots, q$$

$$\lambda_i(t_f) = 0, \quad i = q+1, q+2, \ldots, n$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0}$$

– for relatively simple problems, these equations can be solved using standard linear systems analysis methods

– for more difficult problems, the so-called "sweep" method provides a practical solution alternative

Tracking Controller

- In this type of problem, we want to develop an optimal control law that will force the plant to track a desired reference trajectory, $\boldsymbol{r}(t)$, over a specified time interval

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}$$

$$\boldsymbol{y} = C\boldsymbol{x}$$

$$J = \frac{1}{2}\left\{C\boldsymbol{x}(t_f) - \boldsymbol{r}(t_f)\right\}^T P_f \left\{C\boldsymbol{x}(t_f) - \boldsymbol{r}(t_f)\right\}$$

$$+\frac{1}{2}\int_{t_0}^{t_f} \left\{(C\boldsymbol{x} - \boldsymbol{r})^T Q' (C\boldsymbol{x} - \boldsymbol{r}) + \boldsymbol{u}^T R\boldsymbol{u}\right\} dt$$

$$H = \frac{1}{2}\left\{(C\boldsymbol{x} - \boldsymbol{r})^T Q' (C\boldsymbol{x} - \boldsymbol{r}) + \boldsymbol{u}^T R\boldsymbol{u}\right\} + \boldsymbol{\lambda}^T (A\boldsymbol{x} + B\boldsymbol{u})$$

$$\dot{\boldsymbol{\lambda}} = \left(-\frac{\partial H}{\partial \boldsymbol{x}}\right)^T = -C^T Q' C\boldsymbol{x} - A^T\boldsymbol{\lambda} + C^T Q\boldsymbol{r}$$

$$\begin{aligned}\dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} = A\boldsymbol{x} - BR^{-1}B^T\boldsymbol{\lambda}\\ \boldsymbol{u} &= -R^{-1}B^T\boldsymbol{\lambda}\end{aligned}$$

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\boldsymbol{\lambda}}\end{bmatrix} = \begin{bmatrix}A & -BR^{-1}B^T\\ -C^TQ'C & -A^T\end{bmatrix}\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{\lambda}\end{bmatrix} + \begin{bmatrix}0\\ C^TQ'\end{bmatrix}\boldsymbol{r}$$

Tracking Example:

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= 2x_1 - x_2 + u\end{aligned}$$

$$y = x_1$$

$$J = \left(y_f - 1\right)^2 + \int_0^{t_f} \left\{(y-1)^2 + 0.0025u^2\right\} dt$$

$$A = \begin{bmatrix}0 & 1\\ 2 & -1\end{bmatrix}; \quad B = \begin{bmatrix}0\\ 1\end{bmatrix}; \quad C = \begin{bmatrix}1 & 0\end{bmatrix}$$

$$Q' = 2 \quad R = .005 \quad P_f = 2 \quad r(t) = 1$$

$$C^T Q' = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \qquad C^T Q' C = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\dot{P} = -PA - A^T P - C^T Q' C + PBR^{-1} B^T P$$

$\Rightarrow$

$$\begin{aligned} \dot{s}_{11} &= 2\left(100 s_{12}^2 - 2 s_{12} - 1\right) \\ \dot{s}_{12} &= 200 s_{12} s_{22} + s_{12} - s_{11} - 2 s_{22} \\ \dot{s}_{22} &= 200 s_{22}^2 + 2\left(s_{22} - s_{12}\right) \end{aligned}$$

$$S(t_f) = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\boldsymbol{v} = \left(-A^T + PBR^{-1} B^T\right) \boldsymbol{v} + C^T Q' \boldsymbol{r}$$

$\Rightarrow$

$$\begin{aligned} \dot{v}_1 &= (200 s_{12} - 2)\, v_2 + 2 \\ \dot{v}_2 &= (200 s_{22} + 1)\, v_2 - v_1 \end{aligned}$$

$$\boldsymbol{v}(t_f) = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$$

$$\Rightarrow \quad u(t) = -200 \left\{ s_{12} x_1 + s_{22} x_2 + v_2 \right\}$$

(mostly blank)