Lecture 12

Linear Quadratic Regulator (LQR) Control

Prof. Krishna R. Pattipati
Assisted by : Kihoon Choi

Dept. of Electrical and Computer Engineering
University of Connecticut
Contact: krishna@engr.uconn.edu (860) 486-2890

ECE 6095-4121
Dynamic Modeling and Control of Mechatronic Systems

LQR Controllers

1. **Lyapunov Stability Theory**
   - Main theorem for linear systems
2. **Numerical Solution of Lyapunov Equation**
3. **Constructive Application of Lyapunov Theorem**
   - System stabilization
   - Lyapunov (“bang-bang”) controller
   - Examples
4. **Least Squares Optimization**
   - Problem definition
   - Optimization algorithm
   - Discrete Riccati equation
   - Frequency-weighted LQR (Full-state feedback)
   - Properties of optimal control system (robustness, asymptotic properties)
5. **Examples/Applications**
   - k/s^2 , Inverted pendulum
6. **Rate Weighting**
   - Examples
   - Incorporation of time-delay
7. **Mini-max and H_∞ Controller**
   - Mini-max differential game
   - Synthesizing mini-max controllers

Lyapunov Stability Theory - Preliminaries

- A general theory for studying stability of linear and nonlinear systems
  - Developed ~ 1900; advanced in USA ~ 1960.
  - We consider only linear case here.
  - A useful lead-in to optimal control.
- Quadratic forms

$$v(\underline{x}) = \underline{x}'P\,\underline{x} = p_{11}x_1^2 + 2p_{12}x_1x_2 + \cdots + p_{nn}x_n^2$$

is a quadratic form on $\underline{x}$ if P is positive definite.

An n x n matrix P is positive definite (P > 0) if

(i) $\underline{x}'P\,\underline{x} \geq 0$ for any $\underline{x} \in R^n$

(ii) $\underline{x}'P\,\underline{x} = 0$ if and only if $\underline{x} = 0$

(iii) $P = P'$ (i.e., symmetric)

- Some properties of a positive definite (PD) matrix

Useful tests →
1 - all eigenvalues real, > 0 => P^{-1} exists

2 - eigenvectors are orthogonal, $\underline{\xi}_i'\underline{\xi}_j = 0, \; i \neq j$

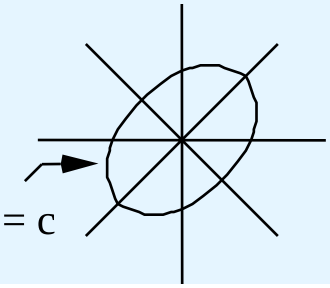
→ 3 - can find S with S'S = P (e.g., Cholesky decomposition) with S invertible

4 - for any $\underline{x}$,

$$0 < \lambda_{min}(P)\,\underline{x}'\underline{x} \;\leq\; \underline{x}'P\,\underline{x} \;\leq\; \lambda_{max}(P)\,\underline{x}'\underline{x}$$

- The equation $\underline{x}'P\underline{x} = c$ defines an ellipsoid in R^n
  - Ellipsoid axes aligned with $\{\underline{\xi}_i\}$
  - Length of semi-major/minor axes = $\sqrt{c / \lambda_i}$

$v(\underline{x}) = \underline{x}'P\,\underline{x} = c$



Application to Stability Analysis

- Study stability of unforced system

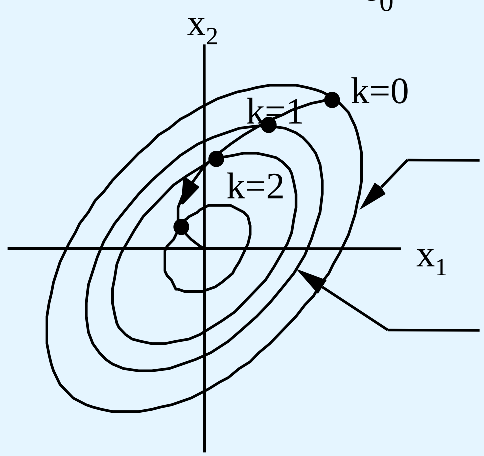
0

$$\underline{x}(k+1) = \Phi\, \underline{x}(k) + \Gamma u(k)$$

$$\underline{x}(0) = \text{initial state}$$

- Suppose we found a quadratic form $v(\underline{x}) = \underline{x}'P\,\underline{x}$ such that when we monitor $v[\underline{x}(k)]$ at any sequence of increasing k:

$$\underbrace{v(\underline{x}(0))}_{c_0} > \underbrace{v(\underline{x}(1))}_{c_1} > \underbrace{v(\underline{x}(2))}_{c_2} > \cdots$$



$v(\underline{x}) = c_0$ = locus of all $\underline{x}$ such that $\underline{x}'P\,\underline{x} = c_0$

=> state is on this contour @ k = 0

$v(\underline{x}) = c_1$ => state lies somewhere on this contour @ k = 1

<u>Implication</u>: $\underline{x}(k) \rightarrow \underline{0}$ as $k \rightarrow \infty$

- Result -

  If we can find a positive scalar (quadratic) function $v(\underline{x})$ such that $v(\underline{x})$ is always decreasing, i.e., if $k_2 > k_1$, $v(\underline{x}(k_2)) < v(\underline{x}(k_1))$ then $\underline{x}(k) \rightarrow \underline{0}$.

  - Such a $v(\underline{x})$ is called a <u>Lyapunov Function</u>.
  - Analogous to a generalized "stored energy".

Main Theorem for Linear Systems

- Existence of a Lyapunov function ==> stability and vice-versa
- Consider $v(\underline{x}) = \underline{x}' P\underline{x}$, $P > 0$, determine

$$\Delta v(\underline{x}) = v(\underline{x}(k+1)) - v(\underline{x}(k))$$

along the system response trajectory $\underline{x}(k+1) = \Phi\, \underline{x}(k)$

$$\Delta v(\underline{x}) = \underline{x}'(k)\Phi' P\Phi\, \underline{x}(k) - \underline{x}'(k)P\underline{x}(k) = -\underline{x}'(k)\underbrace{[P - \Phi' P\Phi]}_{Q}\,\underline{x}(k)$$

- if $Q > 0$, $v(\underline{x}) \downarrow$ and $\underline{x}(k) \rightarrow \underline{0}$. But, if $Q \not> 0$ no conclusions can be drawn.

=> Use reverse procedure. Pick $Q > 0$ and solve

$$P = \Phi' P\Phi + Q \qquad \text{(LEqn)}$$

then

> Theorem: $\underline{x}(k+1) = \Phi\underline{x}(k)$ is stable <u>if and only if</u> given <u>any</u> positive definite Q, the solution P of the equation $P = \Phi' P\Phi + Q$ is positive definite.

- LEqn represents a set of $n(n+1)/2$ linear equations:
  - Expand RHS term $\Phi' P\Phi$
  - Solve for $p_{ij} = p_{ji}$ for $i = 1, \ldots, n$; $j = i, \ldots, n$
  - Solution exists provided $\lambda_i(\Phi)\lambda_j(\Phi) \neq 1$
  - Test if P is positive definite
- A slightly weaker condition is Q positive semidefinite ($Q \geq 0$), as long as $\underline{x}'Q\underline{x} \neq 0$ along a system response trajectory.

Practical Use of Lyapunov Theorem

1. To test stability of Φ, pick a $Q > 0$ and solve for P. If P is not positive definite, system is unstable. If $P > 0$, is stable. Need only do this for one Q.
   Not very practical (there are easier ways to test stability).
   But useful in developing/proving further results . . .
2. If system is stable, Theorem gives an easy way to find a Lyapunov function. Pick any $Q > 0$ (e.g., $Q = \beta I$) and solve LEqn for P. Then $v(\underline{x}) = \underline{x}'P\underline{x}$ is a Lyapunov function, and $\Delta v(\underline{x}) = -\underline{x}'Q\,\underline{x}$. Different Q yield different P.

- Our major efforts will involve finding a $v(\underline{x})$ for a stable system, and using it to develop a SVFB control.

Ex.

Φ, stable (poles at 0.3, 0.4)

$$\underline{x}(k+1) = \begin{bmatrix} 0.2 & -0.2 \\ 0.1 & 0.5 \end{bmatrix} \underline{x}(k)\ ; \quad \text{pick } Q = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Solve $P = \Phi' P \Phi + Q$

$$\begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} = \begin{bmatrix} 0.2 & 0.1 \\ -0.2 & 0.5 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} 0.2 & -0.2 \\ 0.1 & 0.5 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$$= \begin{bmatrix} 0.04p_{11} + 0.04p_{12} + 0.01p_{22} + 2 & -0.04p_{11} + 0.08p_{12} + 0.05p_{22} \\ -0.04p_{11} + 0.08p_{12} + 0.05p_{22} & 0.04p_{11} - 0.2p_{12} + 0.25p_{22} + 2 \end{bmatrix}$$

$$\left.\begin{aligned} 0.96p_{11} - 0.04p_{12} - 0.01p_{22} &= 2 \\ 0.04p_{11} + 0.92p_{12} - 0.05p_{22} &= 0 \\ -0.04p_{11} + 0.2p_{12} + 0.75p22 &= 2 \end{aligned}\right\} \Rightarrow p_{22} = 2.1146.\ p_{12} = 0.0583,\ p_{11} = 2.7639$$

$$P = \begin{bmatrix} 2.1146 & 0.0583 \\ 0.0583 & 2.7639 \end{bmatrix} > 0$$

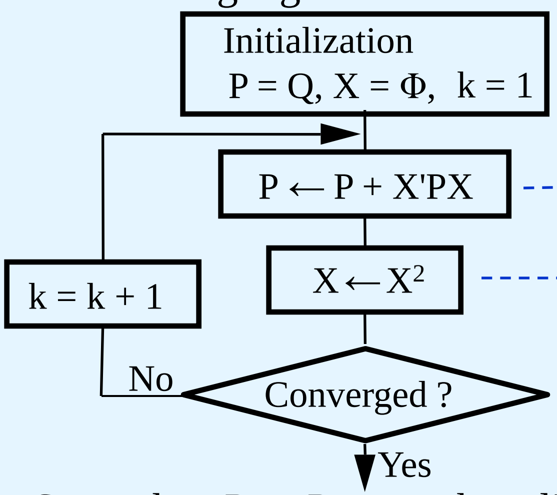
and $\underline{x}'P\underline{x} = p_{11}x_1^2 + 2p_{12}x_1x_2 + p_{22}x_2^2$ is a Lyapunov function.

Numerical Solution of the Lyapunov Equation

- Setting up and solving the n(n+1)/2 system is not practical
  - Requires $0(n^6)$ operations for large n
  - Desire an algorithm requiring $0(n^3)$ operations
- If $|\lambda_i(\Phi)| < 1$, i.e., system is stable, $P = \sum_{i=0}^{\infty} (\Phi')^i Q \Phi^i$ satisfies $P = \Phi' P \Phi + Q$ (check by direct substitution)
  - if system is unstable, sum diverges $\rightarrow \infty$
- An efficient way to sum the series

$$P = Q + \Phi' Q \Phi + (\Phi')^2 Q \Phi^2 + (\Phi')^3 Q \Phi^3 + (\Phi')^4 Q \Phi^4 + \cdots$$

  - Doubling algorithm



$$\begin{cases} \text{1st pass: } Q + \Phi' Q \Phi \\ \text{2nd pass: } Q + \Phi' Q \Phi + (\Phi^2)'[Q + \Phi' Q \Phi]\Phi^2 \\ \text{3rd pass: } Q + \Phi' Q \Phi + \ldots + (\Phi')^7 Q \Phi^7 \end{cases}$$

- After k passes through loop: P = 1st 2^k terms in series

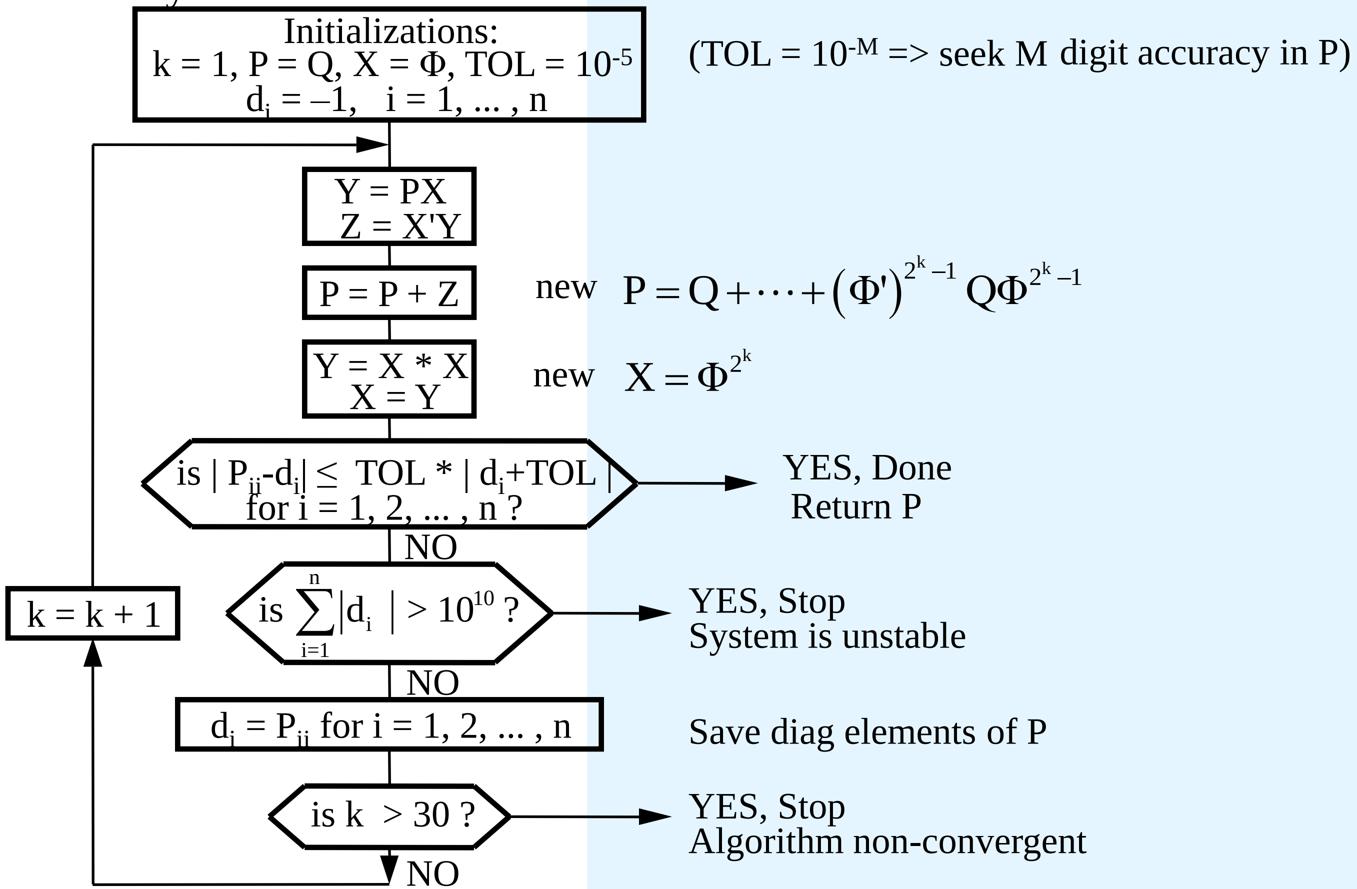
$X = \Phi^{2^k} \rightarrow 0$

- P is monotone increasing
  $P_k = P_{k-1}$ + (a Positive Definite matrix)

- Stop when $P_k \approx P_{k-1}$ or when diagonals $(p_{ii})_k \approx (p_{ii})_{k-1}$ $i = 1, \ldots, n$

Algorithm to Solve Lyapunov Equation, Dlineq

- Imbed a stability test if $P \rightarrow \infty$



- Algorithm generally converges in K ~ 10 iterations
  - Requires ~ K x (2.5) n^3 MADD operations
  - When K = 10 , P has 1024 = 2^{10}, terms, and $\| \Phi^{1024} \| < 10^{-5}$ provided all $|\lambda_i(\Phi)| < 0.99$
- Extremely versatile algorithm

Constructive Application of Lyapunov Theorem to SVFB

(Kleinman, IEEE Transactions AC, June, 1974)

- If $\underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma\underline{u}(k)$ is completely controllable, $\underline{u}(k) = -K_0\underline{x}(k)$ results in a stable closed-loop system where,

  $K_0 = \Gamma' W_M^{-1}\Phi$; M is arbitrary $\geq n$ $\qquad W_M = \sum_{i=0}^{M} \Phi^{-i}\Gamma\Gamma'(\Phi')^{-i} > 0$ via controllability

- Outline of proof (let $\tilde{\Phi} \triangleq \Phi - \Phi\Gamma\Gamma' W_M^{-1}$)

  1. Since $\Phi W_M \Phi' = W_M + \Phi\Gamma\Gamma'\Phi' - \Phi^{-M}\Gamma\Gamma'(\Phi')^{-M}$ show
  $$W_M = \tilde{\Phi} W_M \tilde{\Phi}' + \Phi\Gamma\left[I - \Gamma' W_M^{-1}\Gamma\right]\Gamma'\Phi' + \Phi^{-M}\Gamma\Gamma'(\Phi')^{-M}$$
  2. $I - \Gamma' W_M^{-1}\Gamma = I - \Gamma'\Big[\underbrace{\Gamma\Gamma' + \Phi^{-1} W_{M-1}(\Phi')^{-1}}_{W_M}\Big]^{-1}\Gamma \triangleq Q_1$
  3. Via matrix inversion lemma, $Q_1 = \left[I + \Gamma'\Phi' W_{M-1}^{-1}\Phi\Gamma\right]^{-1} \Rightarrow Q_1 > 0$
  4. Establish that $\underline{x}' W_M \underline{x}$ is a Lyapunov function for $\tilde{\Phi}'$
  $$W_M = \tilde{\Phi} W_M \tilde{\Phi}' + \Phi\Gamma Q_1 \Gamma'\Phi' + \Phi^{-M}\Gamma\Gamma'(\Phi')^{-M}$$
  by showing that $\underline{x}'\Phi\Gamma Q_1\Gamma'\Phi'\underline{x} > 0$ along system response trajectory $\underline{x}(k+1) = \tilde{\Phi}'\underline{x}(k)$ [OK if system is controllable].
  5. By Lyapunov $\tilde{\Phi}'$, and hence $\tilde{\Phi}$, has all eigenvalues with $|\lambda_i(\tilde{\Phi})| < 1$
  6. $\tilde{\Phi} = \Phi(\Phi - \Gamma\underbrace{\Gamma' W_M^{-1}\Phi}_{K_0})\Phi^{-1}$ => $\tilde{\Phi}$ and $\Phi - \Gamma K_0$ have the same eigenvalues

- Corollary: If the system is not completely controllable, $u = -\Gamma' W_M^{\dagger}\Phi\underline{x}$ († denotes pseudo inverse) will stabilize the controllable modes.

Discussion of Stabilization Result: $K_0=\Gamma' W_M^{-1}\Phi$

- Applicable to multi-input systems, $\Gamma = \Psi B \rightarrow$ n x m matrix K_0= m x n gain matrix (m=number of inputs)
- If R = arbitrary m x m positive definite matrix

$$K_0 = -R^{-1}\Gamma' W_{R,M}^{-1}\Phi \text{ is stabilizing}$$

where

$$W_{R,M} = \sum_{i=0}^{M} \Phi^{-i}\Gamma R^{-1}\Gamma'(\Phi')^{-i}$$

  - gives additional degrees of freedom
- Alternate representation

$$K_0 = \left(R + \Gamma' V_{R,M-1}\Gamma\right)^{-1}\Gamma' V_{R,M-1}\Phi$$

$$V_{R,\,M-1} = \Phi' W_{R,\,M-1}^{-1}\Phi$$

- Computing $V_{R,M-1}$

$$V_{R,M-1} = \Phi' W_{R,M-1}^{-1}\Phi = (\Phi')^M \left[\sum_{i=0}^{M-1} \Phi^i \Gamma R^{-1}\Gamma'(\Phi')^i\right]^{-1}\Phi^M$$

1. Pick $M = 2^p \geq n$ (best to pick min p such that $2^p > n$)
2. Go through doubling algorithm p times: $\Phi \rightarrow \Phi'$, $Q = \Gamma R^{-1}\Gamma'$

$$P = \sum_{i=0}^{2^p-1} \Phi^i Q (\Phi')^i \quad ; \quad X = (\Phi')^{2^p}$$

3. Use Cholesky decomposition $P = S'S$, then $V_{R,M-1} = (XS^{-1}) \cdot (XS^{-1})'$

- CL eigenvalues are inside unit circle , but otherwise unspecified.
  - Not a design method for feedback control, rather a starting point.

Examples of System Stabilization, Scalar u, R=1

$$K_0 = (I + \Gamma' V_{M-1} \Gamma)^{-1} \; \Gamma' V_{M-1} \Phi$$

where

$$V_{M-1} = (\Phi')^M \left[\sum_{i=0}^{M-1} \Phi^i \Gamma \Gamma' (\Phi')^i \right]^{-1} \Phi^M$$

- Satellite system (double integrator)

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \xrightarrow{h=1} \Phi = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}; \quad \Gamma = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}$$

$M = 2^2 = 4$

$$V_3 = \begin{bmatrix} 0.20 & 0.40 \\ 0.40 & 1.05 \end{bmatrix}; \quad K_0 = [\; 0.20 \;\; 0.70 \;]$$

CL poles of $\Phi - \Gamma K_0 = 0.6 \pm j0.2$ => $\zeta = 0.82, \omega_n = 0.56$

As M is increased K_0 decreases, and CL poles $\rightarrow 1, 1$

$M = 8$: $K_0 = [\; 0.067 \;\; 0.411 \;]$ => $z_i = 0.78 \pm j0.13; \zeta = 0.82, \omega_n = 0.29$

$M = 16$: $K_0 = [\; 0.02 \;\; 0.225 \;]$ => $z_i = 0.88 \pm j0.076; \zeta = 0.82, \omega_n = 0.15$

Examples of System Stabilization, Scalar u, R=1(Cont'd)

- Inverted pendulum on cart, h = 0.18

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 11 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \end{bmatrix} \rightarrow \Phi = \begin{bmatrix} 1.18 & 0.19 & 0 & 0 \\ 2.10 & 1.18 & 0 & 0 \\ -.017 & -.001 & 1 & .18 \\ -.19 & -.017 & 0 & 0 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} -.017 \\ -.19 \\ .016 \\ .181 \end{bmatrix}$$

Continuous system's open-loop poles @ $0, 0, \pm\sqrt{11}$

Pick $M = 8 = 2^3$: $K_0 = [\ -47.4\ \ -14.7\ \ -6.89\ \ -6.75\ ]$

$$\text{CL poles of } \Phi - \Gamma K_0 = \begin{cases} 0.57 \pm j0.30 & (\zeta = 0.67,\ \omega_n = 3.62) \\ 0.48 \pm j0.065 & (\zeta = 0.98,\ \omega_n = 4.08) \end{cases}$$

<u>with increased M = 12</u>:

$$K_o = [-31.5\ \ -9.67\ \ -1.99\ -2.90] \rightarrow \begin{cases} 0.74 \pm j0.16 & (\zeta = 0.79,\ \omega_n = 2) \\ 0.54 \pm j0.12 & (\zeta = 0.94,\ \omega_n = 3.5) \end{cases}$$

<u>with decreased M = 5</u>:

$$K_o = [-87.9\ \ -26.7\ \ -25.8\ -17.0] \rightarrow \begin{cases} 0.31 \pm j0.39 & (\zeta = 0.61,\ \omega_n = 6.3) \\ 0.34 \pm j0.10 & (\zeta = 0.97,\ \omega_n = 6.0) \end{cases}$$

=> as M increases, K_o decreases and we get slower CL response

Lyapunov Controllers

- Consider a stable SISO system with bounded control

$$\underline{x}(k+1) = \Phi\, \underline{x}(k) + \Gamma u(k); \;\; |\, u(k)\,| \le c_1$$

- Obtain a Lyapunov function $v(\underline{x}) = \underline{x}' P\, \underline{x}$ for free part

$$P = \Phi' P \Phi + Q$$

  Q = arbitrary PD matrix

- Along trajectory of controlled system,

$$\Delta v(\underline{x}) = \underline{x}'(k+1)P\underline{x}(k+1) - \underline{x}'(k)P\underline{x}(k) = \underline{x}'(k)\overbrace{[\Phi' P \Phi - P]}^{-Q}\underline{x}'(k) + 2u(k)\Gamma' P \Phi \underline{x}(k) + u^2(k)\,\Gamma' P \Gamma$$

- Idea: Pick u(k) to drive $\underline{x}(k) \to \underline{0}$ even faster than open-loop.

  Make $\Delta v(\underline{x})$ as negative as possible. Set $\partial\left[\Delta v\left(\underline{x}\right)\right] / \partial u\left(k\right) = 0$:

  - $u(k) = -(\Gamma' P \Gamma)^{-1}\Gamma' P \Phi \underline{x}(k)$ if $|u(k)| < c_1$
  - $u(k) = -c_1 \cdot \text{sgn}\,[(\Gamma' P \Gamma)^{-1}\Gamma' P \Phi \underline{x}(k)]$ if $|u(k)| \ge c_1$

- Algorithm: $K = (\Gamma' P \Gamma)^{-1}\Gamma' P \Phi$
  1. Compute $w = -K\underline{x}(k)$
  2. If $|\, w\,| < c_1$ set $u(k) = w$, else $u(k) = c_1 \text{sgn}(w)$
- As $h \to 0$, $u(k) \to$ "bang-bang" controller; $u(k) = \pm\, c_1$
- Different $Q \to$ different P and K => Q = "design" parameters

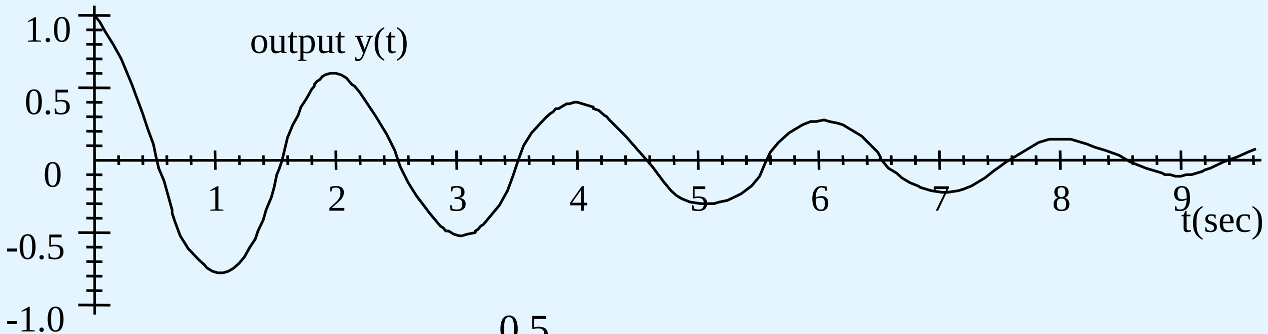
$$\Delta v(\underline{x}) = -\underline{x}'\,[Q + \Phi' P \Gamma(\Gamma' P \Gamma)^{-1}\Gamma' P \Phi]\,\underline{x} \quad \text{in linear region}$$

  - Does increasing q_{ii} speed up the response?, i.e., drive $x_i \to 0$ faster

Example – Lightly Damped System

$\dot{\underline{x}} = \begin{bmatrix} 0 & 1 \\ -10 & -0.5 \end{bmatrix}\underline{x} + \begin{bmatrix} 0 \\ 10 \end{bmatrix}u\,,\;\; y = [\,1\;\;0\,]\underline{x}\,;$ open-loop $\zeta \approx .08,\ \omega_n = \sqrt{10}$; $\underline{x}(0) = [\,1\;\;0\,]'$

- Open-loop system response

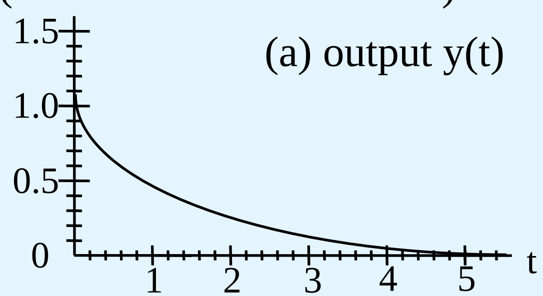


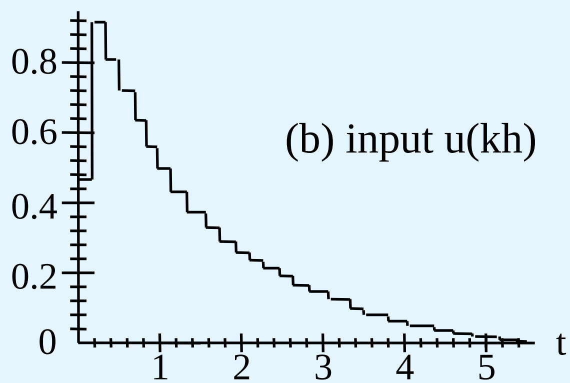
- Lyapunov digital design, $h = \dfrac{0.5}{|\lambda_{max}(A)|} \approx 0.15$

Pick $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow P = \begin{bmatrix} 74.0 & 0.448 \\ 0.448 & 7.85 \end{bmatrix}$

$K = (\Gamma' P \Gamma)^{-1}\Gamma' P \Phi = [\,-0.469\;\;0.631\,]$ => {poles of $\Phi - \Gamma K$ @ 0, 0.887}

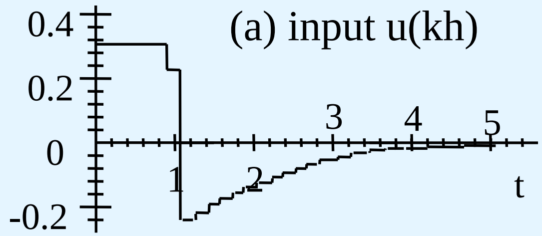
- Response with unconstrained u (dominant 1st order mode)

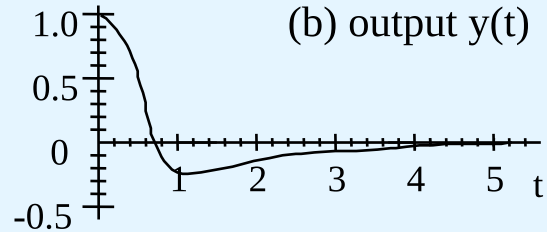




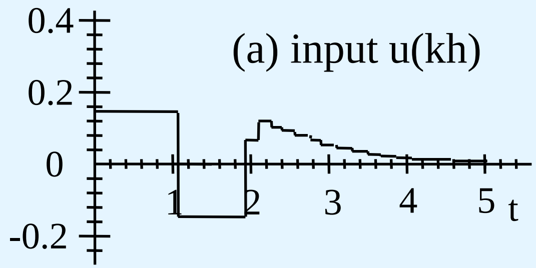
Constrained Response, $\underline{x}(0)=[1 \;\; 0]'$

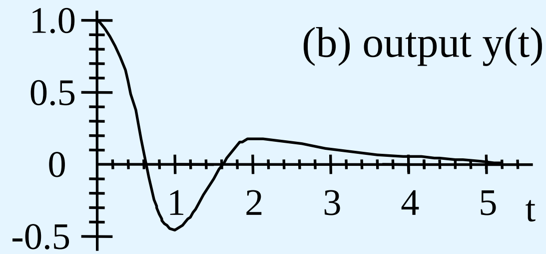
- $|u(k)| \leq 0.30$





- $|u(k)| \leq 0.15$





- "Bang-bang" behavior until $|K\,\underline{x}(k)| \leq c_1$, whereupon closed-loop linear response takes over.
- Modification of q_{11}, q_{22} has small effect on response.

General comments:

- System must be open-loop stable to compute P
- Very little control over time response or poles
- Lyapunov controller useful in cases where h ~ large
- Applicable to multi-input case, $u_i = -c_i \,\text{sat}[K_i\underline{x}(k)/c_i]$
- Assures CL system stability even when control is limited – not necessarily true in other SVFB design approaches.

Introduction to Least Squares Optimization

$\underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma\underline{u}(k)$; $\underline{x}(0)$ = initial state

u(k) = unconstrained

- Objective: Determine a SVFB control $\underline{u}(k) = -K\underline{x}(k)$ so that $\underline{x}(k) \to \underline{0}$
  "nicely" => stability and more
- Pole placement approach
  - Don't necessarily know where good $\lambda_i(\bar{\Phi})$ pole locations are
  - Resulting system may have low | RD | or ϕ_m
  - Needed gains often too big ==> need to manage $|\underline{u}(k)|$
- Optimal control approach
  - Express mathematically what you are trying to achieve

(1) Each $x_i(k) \to 0$ nicely: consider minimizing

$$ISE = \sum_{k=0}^{\infty} q_{11}x_1^2(k) + \cdots + q_{nn}x_n^2(k) = \sum_{k=0}^{\infty} \underline{x}^T(k)Q\underline{x}(k)$$

q_{ii} ~ scale factors to weight relative importance of different errors, $q_{ii} \geq 0$

Q = positive (semi) definite, usually diagonal

(2) Don't want u(k) to be too large: conserve energy $E = \sum_{k=0}^{\infty} \underline{u}^T(k)R\underline{u}(k)$

(3) Combine into a composite criterion, J = ISE + E

$$J = \sum_{k=0}^{\infty}\left[\underline{x}^T(k)Q\underline{x}(k) + \underline{u}^T(k)R\underline{u}(k)\right]$$

R > 0 adjusts tradeoff between speed of response and magnitude of control input.

General Comments

- Linear quadratic (LQ) optimal control problem

  Find u(k) = – K$\underline{x}$(k) to minimize $J = \sum_{k=0}^{\infty} \left[\underline{x}^T(k) Q \underline{x}(k) + \underline{u}^T(k) R \underline{u}(k) \right]; Q \geq 0; R > 0$

- General quadratic cost functional
  - Historical use (from Gauss, Wiener, Kalman, etc.)
  - Physical appeal: larger deviations from nominal are weighted more heavily
  - Physical interpretation: energy is generally ~ x_i^2, u_i^2
  - Mathematically tractable ("easy" to take $\partial / \partial K$)
  - Most overworked problem in modern control theory

- Properties of J
  - J ≥ 0, zero only if $\underline{x}(0)$ is such that free response satisfies $\underline{x}'(0)(\Phi')^k Q (\Phi)^k \underline{x}(0) \equiv 0$
  - Any feedback control that gives a finite value to J must necessarily be stabilizing
  - If R =0, "optimal" control would try to place CL poles at $z_i \to 0$ (drive $\underline{x}(0) \to \underline{0}$ as fast as possible)

<u>Special case</u>: if only concerned about output deviations, consider minimizing

$$J = \sum_{k=0}^{\infty} \underline{y}^T(k) Q_1 \underline{y}(k) + \underline{u}^T(k) R \underline{u}(k)$$

since y(k) = C $\underline{x}$(k), $J = \sum_{k=0}^{\infty} \underline{x}^T(k) \underbrace{\left[C^T Q_1 C \right]}_{Q} \underline{x}(k) + \underline{u}^T(k) R \underline{u}(k)$

=> a "special" case of general state weightings

Optimization Approach

- An expression for J
  - Let $\underline{u}(k) = -K\underline{x}(k)$ be any FB control such that CL system $\underline{x}(k+1) = (\Phi - \Gamma K)\,\underline{x}(k)$, is stable, then

$$\underline{x}(k) = (\Phi - \Gamma K)^k \underline{x}(0), \quad \underline{u}(k) = -K\underline{x}(k)$$

$$J = \underline{x}'(0)\underbrace{\left[\sum_{k=0}^{\infty}(\Phi-\Gamma K)'^{k}\,(Q+K'RK)(\Phi-\Gamma K)^{k}\right]}_{P_k}\underline{x}(0)$$

  - Since CL system is stable,

    (1) P_k satisfies the linear (Lyapunov) equation

$$P_k = (\Phi - \Gamma K)' P_k (\Phi - \Gamma K) + Q + K'RK$$

    (2) P_k is positive (semi) definite symmetric

- P_k is called the <u>cost matrix</u> associated with gain K

$$J = \underline{x}'(0)\, P_k\, \underline{x}(0) \text{ for any } \underline{x}(0)$$

  - P_k does not depend on $\underline{x}(0)$ but only on feedback gains K, $P_k \leftrightarrow K$

- Design approach
  - Find the gain K* that gives the "smallest" cost matrix in a positive defines sense, i.e., if $K^* \leftrightarrow P_{k^*} \triangleq P^*$ then for any K with $K \leftrightarrow P_k$

$$\underline{x}'P^*\underline{x} \le \underline{x}'P_k\underline{x} \quad \text{for all } \underline{x}$$

  - Develop an iterative approach to find K*. Start with gain $K_0 \leftrightarrow P_0$, try to find a gain $K_1 \leftrightarrow P_1$ so that $P_1 < P_0$, ie., K_1 is “more optimal” than K_0.

Method for Obtaining K_1 from P_0

- Start with a stabilizing gain $K_0 \leftrightarrow P_0$

$$P_0 = (\Phi - \Gamma K_0)'P_0(\Phi - \Gamma K_0) + Q + K_0'RK_0$$

  - if $K_1 \leftrightarrow P_1$ (assuming K_1 is stabilizing)

$$P_1 = (\Phi - \Gamma K_1)'P_1(\Phi - \Gamma K_1) + Q + K_1'RK_1$$

- Difference $\delta P = P_0 - P_1$ satisfies

$$\delta P = (\Phi - \Gamma K_1)'\delta P(\Phi - \Gamma K_1) + (K_0 - K_1)'(R+\Gamma'P_0\Gamma)(K_0 - K_1)$$
$$+ (K_0 - K_1)'[(R+\Gamma'P_0\Gamma)K_1 - \Gamma'P_0\Phi] + [(R+\Gamma'P_0\Gamma)K_1 - \Gamma'P_0\Phi]'(K_0 - K_1)$$

=> if select

$$K_1 = (R + \Gamma'P_0\Gamma)^{-1}\Gamma'P_0\Phi$$

then by Lyapunov (if the CL matrix $\Phi - \Gamma K_1$ is stable): $\delta P > 0$;
i.e., $P_1 < P_0$ ($\underline{x}'P_1\underline{x} \le \underline{x}'P_0\underline{x}$), so K_1 is "better" than K_0.

- If K_1 is selected as shown $\Phi - \Gamma K_1$ is stable
  - Rewrite equation for P_0

$$P_0 = (\Phi - \Gamma K_1)'P_0(\Phi - \Gamma K_1) + Q + K_1'RK_1 + (HK_0 - \Gamma'P_0\Phi)'H^{-1}(HK_0 - \Gamma'P_0\Phi);\quad H \triangleq R + \Gamma'P_0\Gamma$$

  - Since $Q_{eff} > 0$ and $P_0 > 0$ => $\Phi - \Gamma K_1$ is stable by Lyapunov
- Continue the process! $K_1 \rightarrow P_1 \rightarrow K_2 \rightarrow P_2 \rightarrow \cdots$
  - Each $P_i < P_{i-1}$ => $\{P_i\}$ converge to P^*
    $\{K_i\}$ converge to K^*
  - Each P_i is positive (semi) definite
  - No $\tilde{P}$ can be $< P^*$ => K^* is unique

The Discrete Riccati Equation

- Main algorithm to find optimal gains
  - Select any K_0 such that $\Phi - \Gamma K_0$ is stable
  - then $K^* = \lim_{i\to\infty} K_i$ = optimal gain

    where $K_{i+1} = (R + \Gamma^T P_i \Gamma)^{-1}\Gamma^T P_i \Phi$; $i = 0, 1, \ldots$

  and P_i is the cost matrix associated with gain K_i

  $$P_i = (\Phi - \Gamma K_i)^T P_i (\Phi - \Gamma K_i) + Q + K_i^T R K_i$$

- At convergence:

  $$K^* = (R + \Gamma^T P^* \Gamma)^{-1}\Gamma^T P^* \Phi$$

  $$J_{min} = \underline{x}'(0) P^* \underline{x}(0)$$

  and $P^* = (\Phi - \Gamma K^*)^T P^* (\Phi - \Gamma K^*) + Q + K^{*T} R K^*$

  $$P^* = \Phi' [P^* - P^*\Gamma(R + \Gamma^T P^* \Gamma)^{-1}\Gamma^T P^*]\Phi + Q$$

  $$P^* = \Phi^T P^* (I + \Gamma R^{-1}\Gamma^T P^*)^{-1}\Phi + Q = \Phi^T (P^{*-1} + \Gamma R^{-1}\Gamma^T)^{-1}\Phi + Q$$

Referred to as the "discrete algebraic Riccati equation" (DARE)

  - Alternate schemes, besides the iterative one, exist for solving the DARE directly.
  - P^* = the <u>unique positive definite</u> solution of the DARE.

- Computing P^* via the iterative algorithm
  - Requires only a subroutine to solve Lyapunov equation
  - $K_i \to K^*$ quadratically, $\| K_{i+1} - K^* \| < c \| K_i - K^* \|^2$
  - Convergence typically in ~ 10 iterations {depends upon how close $|\lambda_{max}(\Phi - \Gamma K_i)|$ are to 1}
  - If desire N digit accuracy in P^*, need to solve Lyapunov equation to N+1 digit accuracy
  - Use stabilization algorithm to obtain K_0

Cross-weighted and Continuous Cost Functionals

K^* = m x n optimal FB gain matrix

Two input (control & disturbances)
-Two output (error and measured output)

- Cross-weights in cost functional (M = n x m)

$$J = \sum_{k=0}^{\infty}\left[\underline{x}^T(k)Q\underline{x}(k) + 2\underline{x}^T(k)M\underline{u}(k) + \underline{u}^T(k)R\underline{u}(k)\right]$$

  - Usually arises when weighting a "generalized" output (or error function in TITO formulation)

$$\underline{y}(k) = F\underline{x}(k) + D\underline{u}(k)$$

  - Optimal control is: $\underline{u}(k) = -(R + \Gamma^T \tilde{P}^*\Gamma)^{-1}\Gamma^T\tilde{P}^*\tilde{\Phi}\underline{x}(k) - R^{-1}M^T\underline{x}(k)$

    where $\tilde{P}^*$ satisfies DARE

$$\tilde{P}^* = \tilde{\Phi}^T[\tilde{P}^* - \tilde{P}^*\Gamma(R+\Gamma^T\tilde{P}^*\Gamma)^{-1}\Gamma'\tilde{P}^*]\tilde{\Phi} + \tilde{Q}$$

$$\tilde{\Phi} = \Phi - \Gamma R^{-1}M^T ; \quad \tilde{Q} = Q - MR^{-1}M^T \geq 0$$

- Translation of continuous cost functional

$$J_c = \int_0^{\infty}\left[\underline{x}^T(t)Q_1\underline{x}(t) + \underline{u}^T(t)R_1\underline{u}(t)\right]dt = \sum_{k=\infty}^{\infty}\left[\underline{x}^T(k)Q\underline{x}(k) + 2\underline{x}^T(k)M\underline{u}(k) + \underline{u}^T(k)R\underline{u}(k)\right]$$

$$Q = \int_0^h e^{A^T\sigma}Q_1e^{A\sigma}d\sigma \sim \frac{h}{2}\left[\Phi^TQ_1\Phi + Q_1\right] \qquad M = \int_0^h e^{A^T\sigma}Q_1\int_0^h e^{A\xi}Bd\xi d\sigma \sim \frac{h}{2}\Phi'Q_1\Gamma$$

$$R = hR_1 + \int_0^h\left[\int_0^\sigma e^{A\xi}Bd\xi\right]^TQ_1\left[\int_0^\sigma e^{A\xi}Bd\xi\right]d\sigma \sim hR_1 + \frac{h}{2}\Gamma'Q_1\Gamma$$

(easier to use gain equivalence $K^*|_{\text{continuous}} \rightarrow \tilde{K}^*|_{\text{discrete}}$)

LQR with Frequency Weighted Cost Functional

- Recall Parseval's theorem for discrete-time systems (one-sided)

$$\sum_{k=0}^{\infty} \underline{g}^T(k)\underline{g}(k) = \frac{h}{\pi}\int_0^{\pi/h} \underline{G}^T(e^{-j\omega h})\underline{G}(e^{j\omega h})d\omega$$

- LQR with frequency weighted cost functional

$$\mathrm{J_c} = \frac{h}{\pi}\int_0^{\pi/h}[\underline{y}^T(e^{-j\omega h})W_1^T(e^{-j\omega h})W_1(e^{j\omega h})\underline{y}(e^{j\omega h}) + \underline{u}^T(e^{-j\omega h})W_2^T(e^{-j\omega h})W_2(e^{j\omega h})\underline{u}(e^{j\omega h})]d\omega$$

$W_1(e^{j\omega h})$ and $W_2(e^{j\omega h})$ are frequency weights

Typically,
$W_1(z) = w_1(z)I_p$
$W_2(z) = w_2(z)I_m$

- Transform the cost functional back to time domain via the following steps:

Represent $\underline{y}_1(z) = W_1(z)\underline{y}(z) \Rightarrow W_1(z) = C_1(zI - \Phi_1)^{-1}\Gamma_1 + D_1$

Similarly, $\underline{y}_2(z) = W_2(z)\underline{u}(z) \Rightarrow W_2(z) = C_2(zI - \Phi_2)^{-1}\Gamma_2 + D_2$

$$\Rightarrow \mathrm{J_c} = \frac{h}{\pi}\int_0^{\pi/h}[\underline{y}^T(e^{-j\omega h})W_1^T(e^{-j\omega h})W_1(e^{j\omega h})\underline{y}(e^{j\omega h}) + \underline{u}^T(e^{-j\omega h})W_2^T(e^{-j\omega h})W_2(e^{j\omega h})\underline{u}(e^{j\omega h})]d\omega$$

$$= \sum_{k=0}^{\infty}[\underline{y}_1^T(k)\underline{y}_1(k) + \underline{y}_2^T(k)\underline{y}_2(k)]$$

- Augmented system

$$\begin{bmatrix} \underline{x}(k+1) \\ \underline{x}_1(k+1) \\ \underline{x}_2(k+1) \end{bmatrix} = \begin{bmatrix} \Phi & 0 & 0 \\ \Gamma_1 C & \Phi_1 & 0 \\ 0 & 0 & \Phi_2 \end{bmatrix}\begin{bmatrix} \underline{x}(k) \\ \underline{x}_1(k) \\ \underline{x}_2(k) \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \\ \Gamma_2 \end{bmatrix}\underline{u}(k) \Rightarrow \underline{X}_a(k+1) = \Phi_a\underline{X}_a(k) + \Gamma_a\underline{u}(k)$$

$$\begin{bmatrix} \underline{y}_1(k) \\ \underline{y}_2(k) \end{bmatrix} = \begin{bmatrix} D_1 C & C_1 & 0 \\ 0 & 0 & C_2 \end{bmatrix}\begin{bmatrix} \underline{x}(k) \\ \underline{x}_1(k) \\ \underline{x}_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ D_2 \end{bmatrix}\underline{u}(k) \Rightarrow \underline{Y}_a(k) = C_a\underline{X}_a(k) + D_a\underline{u}(k)$$

Cross-weighted cost functional

$$\mathrm{J_c} = \sum_{k=0}^{\infty}\underline{Y}_a^T(k)\underline{Y}_a(k)$$

$$= \sum_{k=0}^{\infty}[\underline{X}_a^T(k)C_a^T C_a\underline{X}_a(k) + 2\underline{X}_a^T(k)C_a^T D_a\underline{u}(k) + \underline{u}^T(k)D_a^T D_a\underline{u}(k)]$$

Dynamic compensator:

$$\underline{u}(k) = -K_a\underline{X}_a(k) = -K_x\underline{x}(k) - K_1\underline{x}_1(k) - K_2\underline{x}_2(k)$$

Application of the Optimal Control

- We can show $\underline{u}(k) = -K^* \underline{x}(k)$ is the optimal control, not just the linear optimal one.
- The closed-loop $\underline{x}(k+1) = \Phi \underline{x}(k) - \Gamma K^* \underline{x}(k)$ must be stable.
- Selection of weightings
  - Major design step in method's application
  - Initial design:

$$q_{ii} = \text{relative weighting on state } x_i = \frac{1}{|x_{i,max}|^2}$$

where $x_{i,max}$ = maximum desired (or anticipated) value of $x_i(k)$. If unconcerned about x_i deviations from zero, set $q_{ii} = 0$.

  - Adjust control weighting r_{ii} to achieve desired balance between control usage and response speed. Initially,

$$r_{ii} = \frac{1}{|u_{imax}|^2}$$

  - "Tune" q_{ii}, r_{ii} to obtain desired CL time response starting with representative $\underline{x}(0)$s

=> increase q_{jj} to decrease RMS x_j decrease r_{ii} to increase CL speed of response trade-off errors in $x_j \leftrightarrow x_i$ via q_{jj} vs. q_{ii}

- Basically, approach is time-domain oriented, but
  - Examine CL pole locations, ϕ_m, ω_c, etc.
- Other "techniques" and "rules" exist for picking weights.

Properties of the Optimal CL system - 1

(1) Closed-loop pole locations

- Closed-loop poles are the n roots inside unit circle of

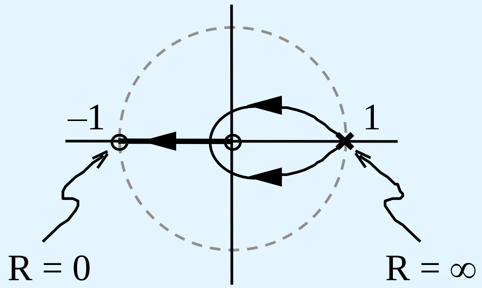
$$\det\ [\ R + \Gamma'(z^{-1}I - \Phi')^{-1}Q(zI - \Phi)^{-1}\Gamma\ ] = 0$$

  - In single input case, if Q = C'C (output weighting only), closed-loop poles satisfy

$$R + \tilde{G}(z^{-1})\,\tilde{G}(z) = 0$$

  => optimal CL poles of $\Phi - \Gamma K^*$ are not arbitrary

- Example: Satellite system, $\tilde{G}(z) = \frac{1}{2}\frac{(z+1)}{(z-1)^2}$, output weighting only



Root locus of CL poles R: $\infty \rightarrow 0$

$$1 + \frac{1}{4R}\frac{z(z+1)^2}{(z-1)^4} = 0$$

(Consider branches with $|z| < 1$ only)

  - As $R \rightarrow 0$, CL poles follow a locus of constant damping $\zeta = .707$, until $R = R_0 = 0.025$. Then, for $R < R_0$ have 2 real roots on (–1, 0)!

  => too small a value of R will give oscillatory CL response.

- General property as $R \rightarrow 0$: (single input case with Q = C'C)
  - Assume $\tilde{G}(z)$ has r zeros $\delta_1, \delta_2, ..., \delta_r$
  - As $R \rightarrow 0$, r closed-loop poles $\rightarrow$ r zeros of $\tilde{G}(z)\tilde{G}(z^{-1})$ inside or on unit circle. The remaining n – r poles $\rightarrow z = 0$. (in ex. $r = 1$, $\delta_1 = -1$)
  - i.e., if δ_i is a zero of $\tilde{G}(z)$, a CL pole $\rightarrow \delta_1$ or $1/\delta_i$ (whichever has magnitude < 1) as $R \rightarrow 0$.

Properties of the Optimal CL system - 2

(2) Return difference and phase margin

- Loop gain properties, $LG(z) = K^*(zI-\Phi)^{-1}\Gamma$
  - via algebraic manipulations on Riccati equation:

$$[I + LG(z^{-1})]'(R + \Gamma'P^*\Gamma)[I + LG(z)] = R + \Gamma'(z^{-1}I - \Phi')^{-1}Q(zI - \Phi)^{-1}\Gamma$$

  - in single input case, factor $Q = S'S$

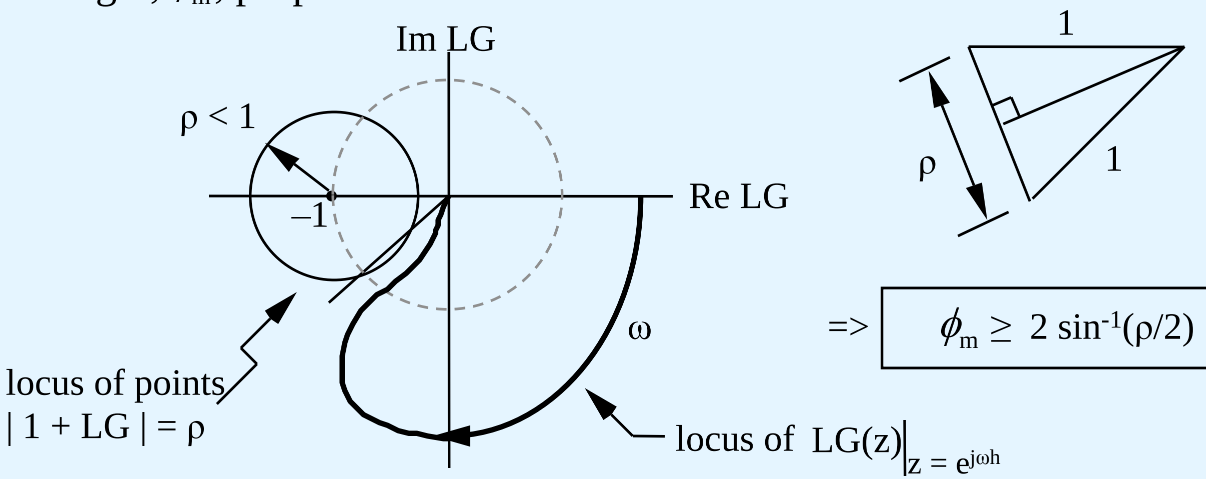
$$\tilde{G}_{eff}(z) \triangleq S(zI - \Phi)^{-1}\Gamma$$

$[\tilde{G}_{eff}(z) = \tilde{G}(z) \text{ if } Q = C^T C]$

$$|1 + LG(z)|^2 = \frac{R + \tilde{G}'_{eff}(z^{-1})\tilde{G}_{eff}(z)}{R + \Gamma'P^*\Gamma} \geq \frac{R}{R + \Gamma'P^*\Gamma} = \rho^2 \quad (\rho < 1)$$

$$\Rightarrow \ |\text{Return difference}| = |1 + K^*(zI - \Phi)^{-1}\Gamma|\Big|_{z = e^{j\omega h}} \geq \rho$$

- Phase margin, ϕ_m, properties



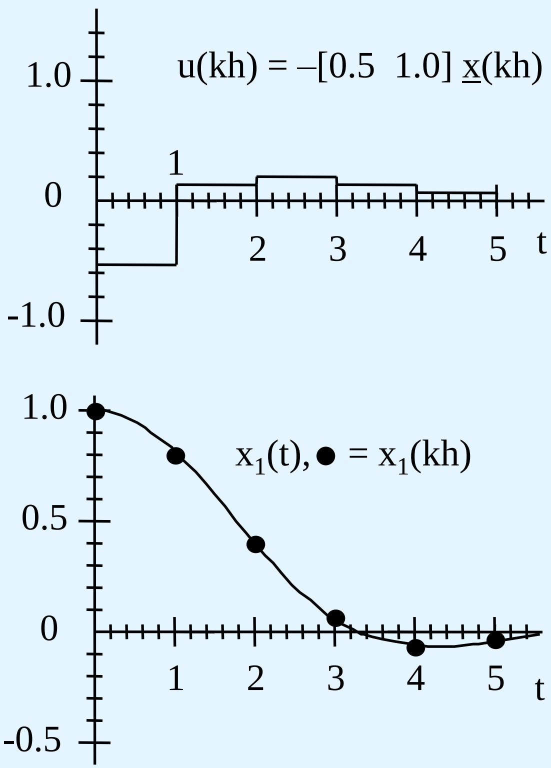
$$\Rightarrow \quad \phi_m \geq 2\sin^{-1}(\rho/2)$$

Example – Satellite (Double Integral) System, h=1

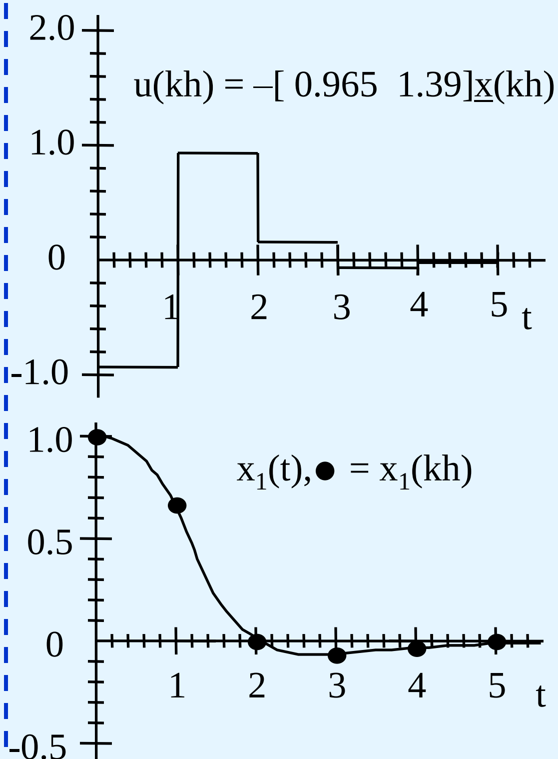
$$\underline{x}(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \underline{x}(k) + \begin{bmatrix} 0.5 \\ 1.0 \end{bmatrix} \underline{u}(k)\ ; \quad y(k) = x_1(k)\ ; \quad \underline{x}(0) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$q_{11} = 1$, $q_{22} = 0$ (interested in output → 0), R = design parameter

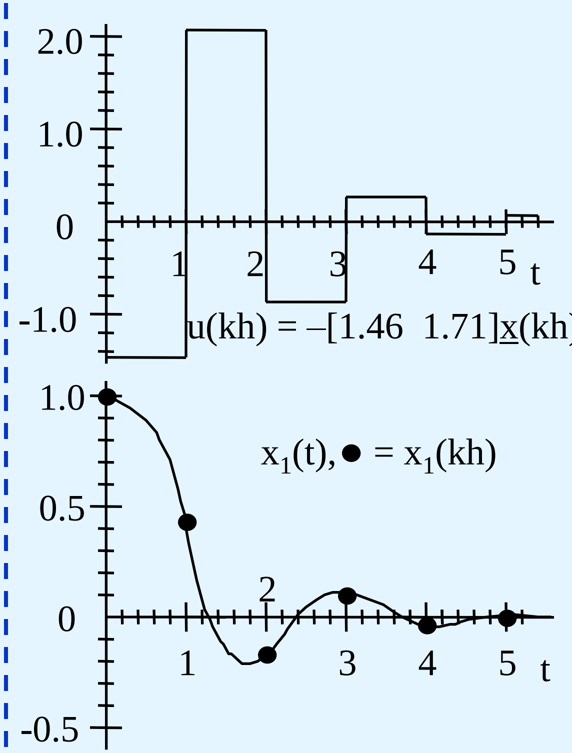
- R = 1 yields sluggish response



- R = 0.1 gives faster response



- R = 0.01 generates CL response with "ripple"



=> Examine CL pole locations as a function of Q, R

Example – Inverted Pendulum on a Cart, h=0.18 sec

$$\underline{x} = \left[\theta,\ \ \dot{\theta},\ \ d,\ \ \dot{d}\right]'$$

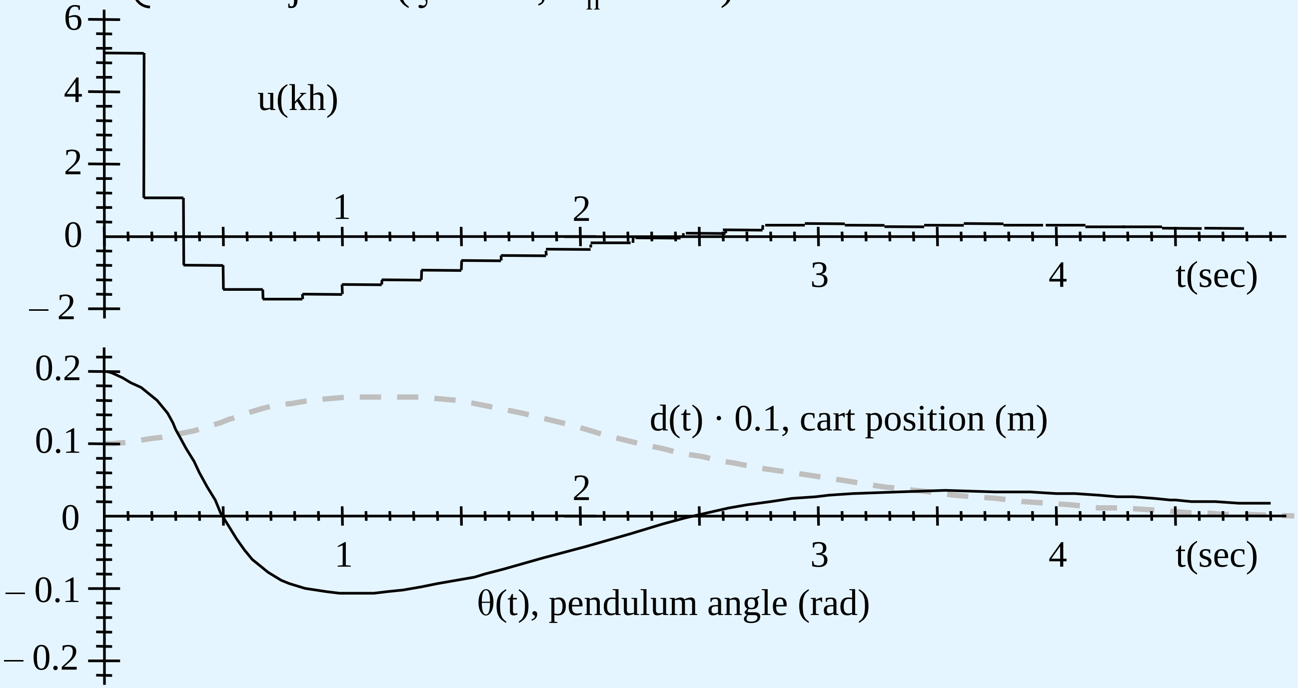
Initial design

$$\left.\begin{array}{l}\theta_{max} \approx 0.5 \text{ rad} \\ d_{max} \approx 1 \text{ meter}\end{array}\right\} \Rightarrow Q = \text{diag}\,[\,4\ \ 0\ \ 1\ \ 0\,]$$

Select R = 1 => $K^* = [\,-22.9\ \ -6.98\ \ -0.487\ \ -1.08\,]$

$$\text{CL poles} = \begin{cases} 0.55 \pm j0.03 \ \ (\zeta \approx 1,\ \omega_n = 3.3) \\ 0.88 \pm j0.11 \ \ (\zeta \approx 0.7,\ \omega_n = 0.95)\end{cases}$$

$\rho = 0.487 \Rightarrow \phi_m \geq 28.2^o$



- Reduce weighting on u to speed response $\theta(t) \rightarrow 0$ (will require more control input).

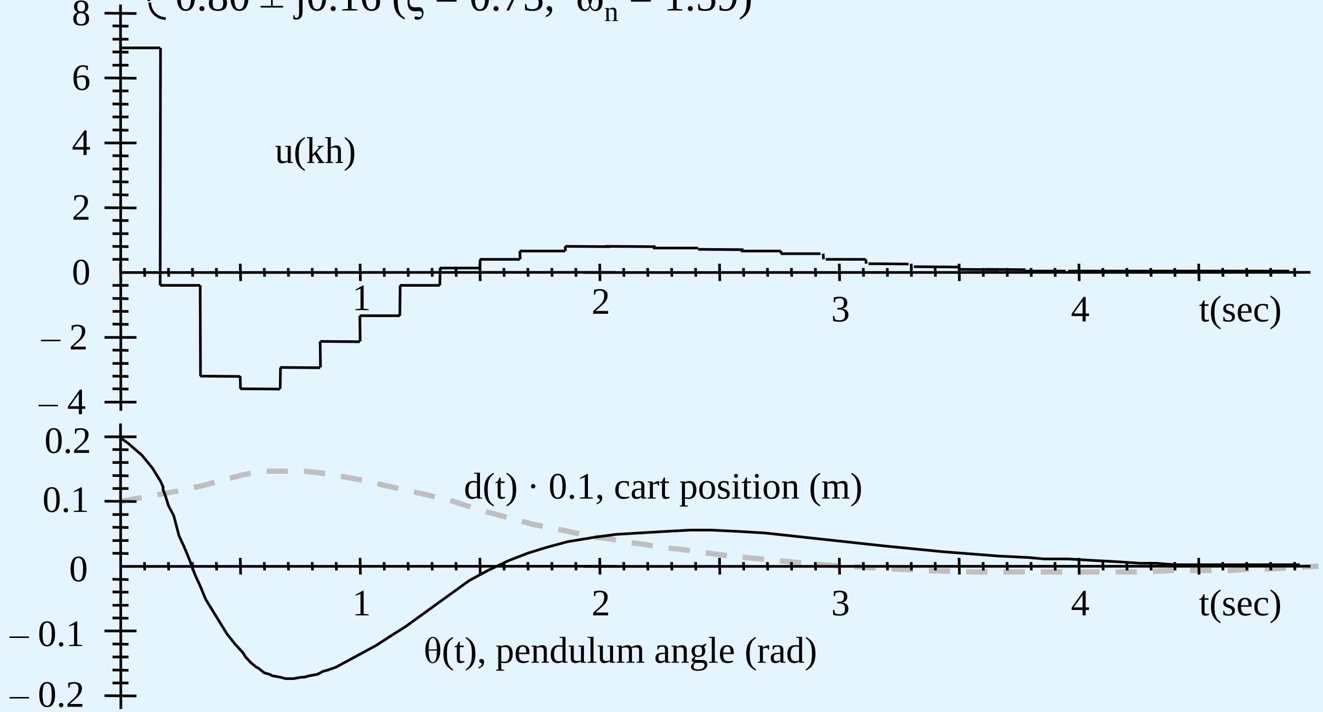
Inverted Pendulum II

2nd Design iteration, R = 0.1

$$K^* = [\,-28.4 \quad -8.67 \quad -1.39 \quad -2.20\,]$$

$$\text{CL poles} = \begin{cases} 0.53 \pm j0.09\ (\zeta = 0.96,\ \omega_n = 3.5) \\ 0.80 \pm j0.16\ (\zeta = 0.73,\ \omega_n = 1.59) \end{cases}$$

$\rho = 0.439 \Rightarrow \phi_m \geq 25.4^o$

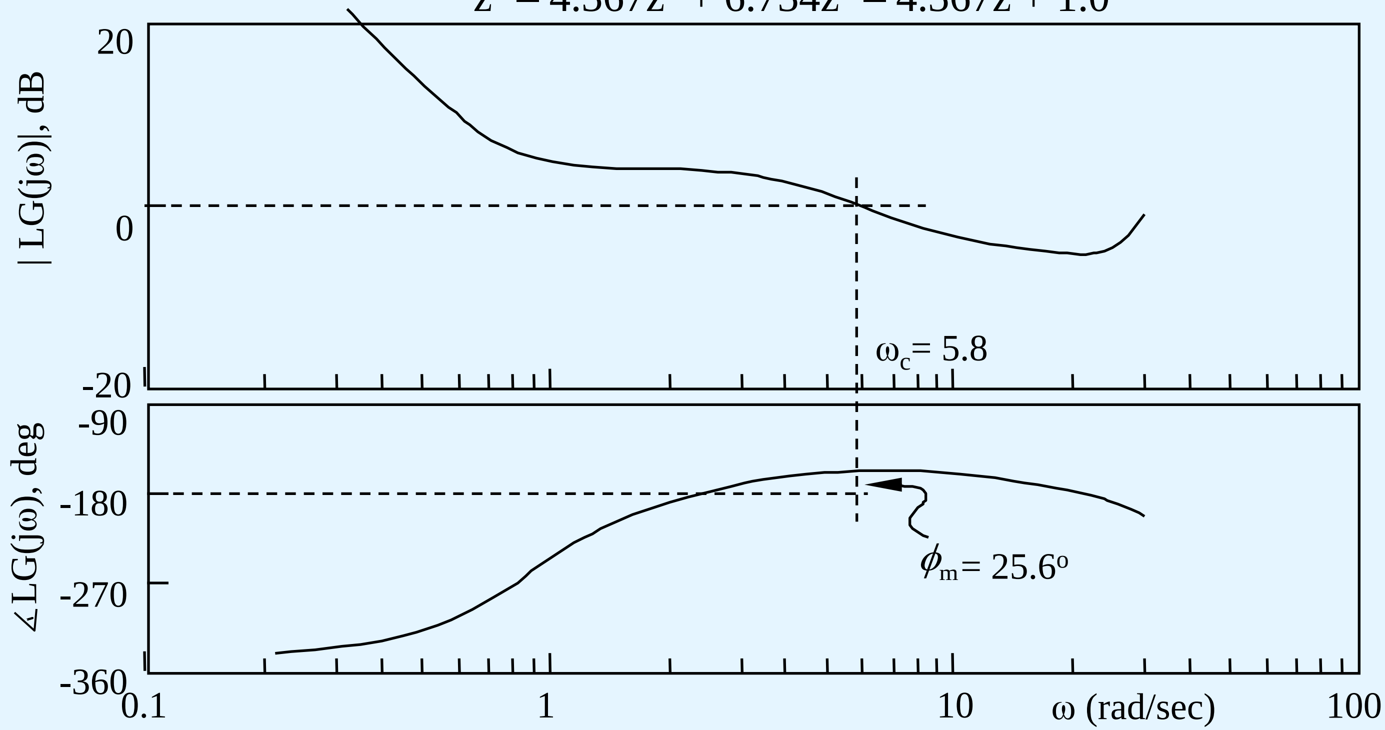


- Further possibilities
  - Further decrease R (e.g., R = .01 yields ~ 3 sec setting time with 1½ → 2 times the amount of control)
  - Modify θ:d = 0.5:1 ratio (minor effect)
  - As $R \rightarrow 0$: $\phi_m \downarrow$, $|u(kh)| \uparrow$, $t_s \downarrow$ and $\theta(t)$ overshoot $\uparrow$

Inverted Pendulum, Phase Margin Analysis

- R = 0.1

$$LG(z) = K^*(zI - \Phi)^{-1}\Gamma = \frac{1.708z^3 - 4.084z^2 + 3.197z - 0.8066}{z^4 - 4.367z^3 + 6.734z^2 - 4.367z + 1.0}$$



  - Formula $\phi_m \geq 2\sin^{-1}(\rho/2)$ reasonably tight (25.4 vs. 25.6 !)
  - As $h \rightarrow 0$, $\rho \rightarrow 1$ and $\phi_m \geq 60^o$ as for optimal continuous design
  - Generally as gains increase, ϕ_m decreases
- For single input systems, optimal control design…pole placement design with the same poles.
  - But note that pole placement can achieve CL pole locations where an optimal design will not/can not

Summary of Optimal Control Design Method

- Basically a "smart" pole placement SVFB design
  - SVFB does not modify system zeros
- Based on minimizing a quadratic criterion
  - Function of state and control deviations

=> Advantages

- Straightforward design methodology
- Design parameters (Q, R) relate to CL response
- Directly applicable to MIMO systems
- Small number of design parameters
- Has a guaranteed lower bound on ϕ_m
- CL system is always stable
- Numerous extensions can/have been done

e.g., integral FB via a small weight on $x_I^2(k) = \left[\sum_{i=1}^{k} x_m(i)\right]^2$

e.g., command following via weighting $[C\underline{x}(k) - r(k)]^2$

=> Disadvantages

- Requires fairly extensive software to do design (dlqr,DARE routines)
- Do not have direct control over CL pole locations (some choices of Q, R can give poles on z < 0)
- Weighting selection process is largely trial and error
- Quadratic criterion not always best
- Need to measure or estimate all states

Weighting of Control Rate

- Usual optimal FB control has high bandwidth
  - Can give problems if actuators are rate-limited
  - Often not necessary if system dynamics are "slow"
- Weight $\Delta(k) = [u(k) - u(k-1)]/h$ in cost functional

$$J = \sum_{k=0}^{\infty}\left[\underline{x}^T(k)Q\underline{x}(k) + Ru^2(k-1) + G\Delta^2(k)\right]$$

- Develop augmented system dynamics, $x_{n+1}(k) = u(k-1)$

$$u(k) = u(k-1) + h\Delta(k)$$

$$\Rightarrow \quad \underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma u(k-1) + h\Gamma\Delta(k)$$

$$x_{n+1}(k+1) = x_{n+1}(k) + h\Delta(k)$$

let $\underline{\chi}(k) = [\,\underline{x}(k), u(k-1)]^T$,

$$\underline{\chi}(k+1) = \underbrace{\begin{bmatrix}\Phi & \Gamma \\ 0 & 1\end{bmatrix}}_{\Phi_a}\underline{\chi}(k) + h\underbrace{\begin{bmatrix}\Gamma \\ 1\end{bmatrix}}_{\Gamma_a}\Delta(k)\,; \quad \underline{\chi}(0) = \begin{bmatrix}\underline{x}(0) \\ 0\end{bmatrix}$$

$$J = \sum_{k=0}^{\infty}\left[\underline{\chi}^T(k)Q_a\underline{\chi}(k) + G\Delta^2(k)\right]$$

Q_a = diag [Q R]

- Solve "augmented" optimal control problem $\underline{\chi}(k) \Leftrightarrow \underline{x}(k)$, $\Delta(k) \Leftrightarrow u(k)$
  - Augmented system is controllable wr to Δ, if original system was controllable wr to u

$$\Delta(k) = -K_a\underline{\chi}(k) = -K_x\underline{x}(k) - K_u u(k-1)$$

  - Alternate structure $u(k) = (1 - hK_u)u(k-1) - hK_x\underline{x}(k)$

Properties of Rate Weighted Controller

$$u(k) = (1-hK_u)u(k-1) - hK_x \underline{x}(k)$$

- Analogous to FB $v = \dot{u} = -k_x \underline{x} - k_u u$ put through a first-order filter $\frac{a}{s+a}$, with $a \sim K_u$
- As $G \to 0$, $K_u \to 1/h$, $K_x \to K^*/h$ and original SVFB control is recovered
- Highly recommended for all physical systems
  - Adds robustness to design
  - Generally gives slightly smaller ω_c
  - Provides ability to manage CL bandwidth
  - Effect trade-off between $\dot{u}$ and u, $\underline{x}$
- Example: Inverted pendulum on a cart, $\underline{x} = \begin{bmatrix} \theta & \dot{\theta} & d & \dot{d} \end{bmatrix}$
  - add a rate weighting to previous design

$$Q = \text{diag}\,[\,4 \;\; 0 \;\; 1 \;\; 0\,],\; R = 0.1$$

$$G = 0.0081 = \left| \frac{h}{\Delta u_{max}} \right|^2 \text{ with } \Delta u_{max} = 2$$

  - FB control with rate weighting

$$u(k) = (1 - hK_u)u(k-1) - hK_x \underline{x}(k)$$

$$K_u = 4.97\;;\; K_x = [-109.5 \;\; -33.4 \;\; -3.61 \;\; -6.29\,]$$

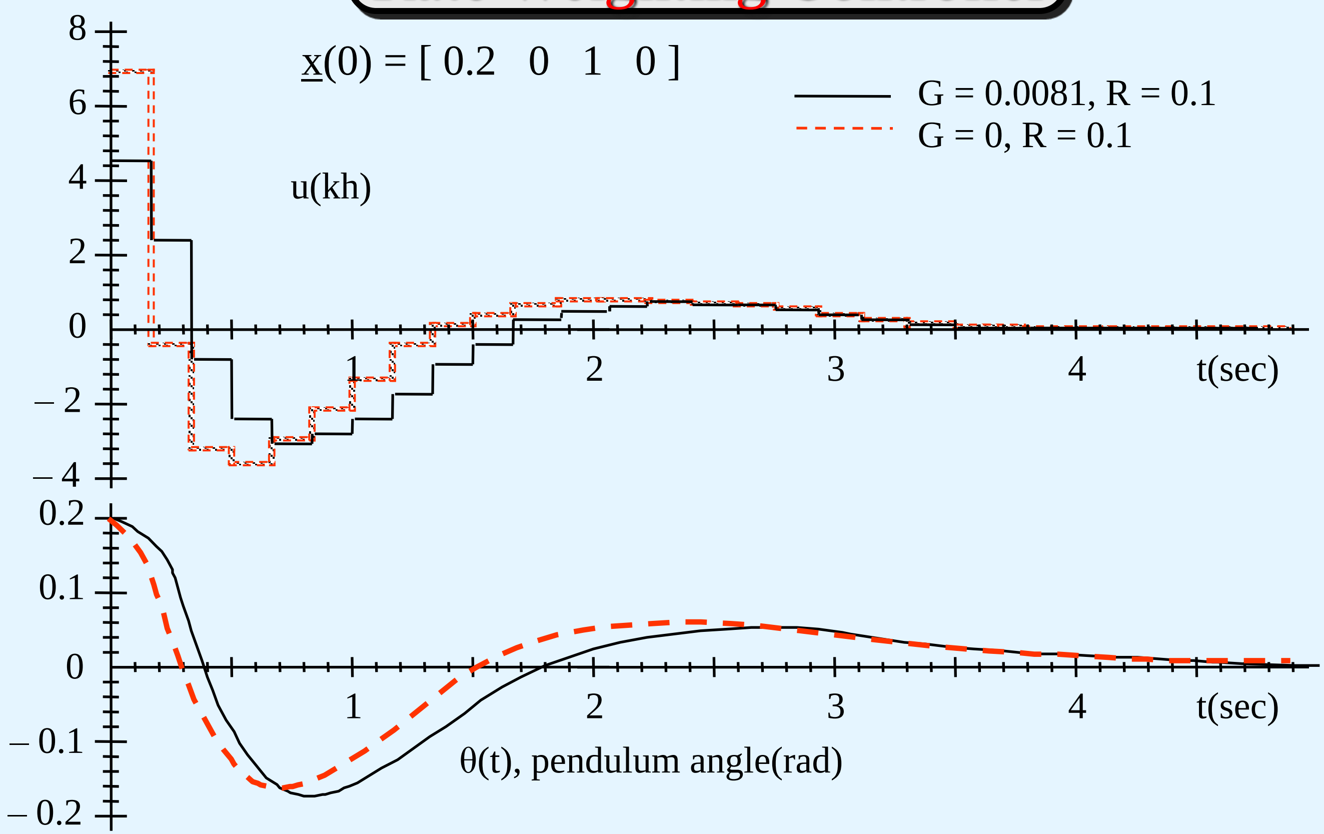
$$\Rightarrow \; K_x/K_u = [-22.0 \;\; -6.72 \;\; -0.726 \;\; -1.27\,]$$

  - By analogy recall K^* for $G = 0$

$$K^* = [-28.4 \;\; -8.67 \;\; -1.39 \;\; -2.20\,]$$

=> gains decrease to compensate for added filtering

Simulation Results – Rate Weighting Controller



- Decrease in magnitude of Δu_{max} (from 7.4 to ≈ 4.5)
  - Accompanied by slower system response, but with ≈ same overshoots
  - Similar to a time scaling effect
- Increasing G to 0.09 further slows response

$$t_s \approx 6 \text{ sec}, \ \ \Delta u_{max} \approx 3.2$$

  - We soon reach a point of diminished return

Compensation for Fractional Time Delay

$$\tau = Mh + \varepsilon\,; \quad M = 0$$

- Recall model for < 1 step (computational) delay

$$\chi(k) \triangleq [\underline{x}(k), u(k-1)]' = \text{augmented state}$$

$$\chi(k+1) = \begin{bmatrix} \Phi & \Gamma_1 \\ 0 & 0 \end{bmatrix} \chi(k) + \begin{bmatrix} \Gamma_0 \\ 1 \end{bmatrix} u(k)$$

  - Can apply optimal control design directly to augmented model when $G = 0$; $Q_a = [Q, \; 0]$.
    [Gives same results as $u(kh) = -K^* \hat{\underline{x}}(kh + \varepsilon)$]
- Alternate time delay model
  - Replace $u(k) \Rightarrow u(k-1) + h\Delta(k)$; note $\Gamma_0 + \Gamma_1 = \Gamma$

$$\chi(k+1) = \begin{bmatrix} \Phi & \Gamma \\ 0 & 1 \end{bmatrix} \chi(k) + h \begin{bmatrix} \Gamma_0 \\ 1 \end{bmatrix} \Delta u(k)$$

  - In desired form for weighting $\Delta(k)$
  - Identical to augmented model but with a modified Γ_a. (When $\varepsilon = h^-$, $\Gamma_0 = 0$.)

$$\Delta(k) = -K_u u(k-1) - K_x \underline{x}(k)$$

  => Natural fit between fractional delay model and weighting of control rate.
  Excellent for $\varepsilon < h$, i.e., compensation of up to one time-step delay.

- For $M \geq 1$ apply state prediction ideas

$$\Delta(k) = -K_u u(k-1) - K_x \hat{\underline{x}}(k + M)$$

$\hat{\underline{x}}(k + M)$ = prediction of $\underline{x}$ at step $k + M$, obtained by propagating

$$\underline{x}(k+1) = \Phi \underline{x}(k) + \Gamma_1 u(k-1-M) + \Gamma_0 u(k-M)$$

Minimax H_∞ Controller - 1

$\underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma\underline{u}(k) + E\,\underline{d}(k)$; $\underline{x}(0)$ = initial state; $\underline{d}(k)$ is unknown but bounded

- Objective: Determine a SVFB control $\underline{u}(k) = -K\underline{x}(k)$ and worst case $\underline{d}(k)$ so that $\underline{x}(k) \to \underline{0}$. It turns out that the worst case $\underline{d}(k) = = -K_d\,\underline{x}(k)$, but we won't feed it back.

$$J = \min_{\underline{u}} \max_{\underline{d}} \sum_{k=0}^{\infty} \left[\underline{x}^T(k) Q \underline{x}(k) + \underline{u}^T(k) R \underline{u}(k) - \gamma^2 \underline{d}^T(k) \underline{d}(k) \right] \sim \textit{Mini}\max\ \textit{criterion}$$

$\Rightarrow$ finds the worst case disturbance if can find smallest $\gamma \Rightarrow H_\infty$-full state feedback controller

- An expression for J assuming $[\Phi - \Gamma K - E K_d]$ is stable . Actually, need $\Phi - \Gamma K$ to be stable

$$J = \sum_{k=0}^{\infty} \underline{x}^T(k)[Q + K^T R K - \gamma^2 K_d^T K_d]\underline{x}(k) = \underline{x}^T(0) P_k \underline{x}(0) = Trace\left(P_k \underline{x}(0)\underline{x}^T(0)\right)$$

where P_k satisfies the Lyapunov equation $P_k = (\Phi - \Gamma K - EK_d)^T P_k (\Phi - \Gamma K - EK_d) + Q + K^T R K - \gamma^2 K_d^T K_d$

- Design approach
  - Find the gains K* and K_d * that optimize the cost matrix in a positive definite sense
  - Following the LQ optimization approach used earlier or Hamiltonian approach next

$$K^* = -R^{-1}\Gamma^T P^*(I_n + SP^*)^{-1}\Phi$$

$$K_d^* = -\frac{1}{\gamma^2} E^T P^*(I_n + SP^*)^{-1}\Phi$$

where $S = \Gamma R^{-1}\Gamma^T - \frac{1}{\gamma^2} E E^T$

P^* is the solution of Discrete Algebraic Riccati Equation:

$$P^* = \Phi^T P^*(I_n + SP)^{-1}\Phi + Q$$
$$= \Phi^T P \Phi - \Phi^T P \Gamma_a (R_a + \Gamma_a^T P \Gamma_a)^{-1} \Gamma_a^T P \Phi + Q$$

where $\Gamma_a = [\Gamma\ E]$ and $R_a = Diag(R, -\gamma^2 I_l)$

- May not have a solution for all $\gamma \Rightarrow$ need to find the range $[\gamma_{min}, \infty]$

Minimax H_∞ Controller - 2

- The closed-loop system matrix Φ - ΓK is stable if [Φ - ΓK-E K_d] is stable.

  Define Lyapunov function $V(\underline{x}(k)) = \underline{x}^T(k)P^*\underline{x}(k)$

  Need to prove $V(\underline{x}(k+1)) - V(\underline{x}(k)) < 0$

  $$know\ \underline{x}^T(k)[(\Phi - \Gamma K - EK_d)^T P^*(\Phi - \Gamma K - EK_d) - P^*]\underline{x}(k) < 0$$

  $$\Rightarrow -\underline{x}^T(k)[Q + K^T RK - \frac{1}{\gamma^2}K_d^T K_d]\underline{x}(k) < 0 \Rightarrow -\underline{x}^T(k)[Q + K^T RK]\underline{x}(k) < 0$$

- Hamiltonian approach

$$\text{Problem}: \min_{\underline{u}} \max_{\underline{d}} \frac{1}{2}\sum_{k=0}^{\infty}\left[\underline{x}^T(k)Q\underline{x}(k) + \underline{u}^T(k)R\underline{u}(k) - \gamma^2\underline{d}^T(k)\underline{d}(k)\right] s.t.\ \underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma\underline{u}(k) + E\underline{d}(k)$$

Define Hamiltonian:

$$H(\underline{x}(k), \underline{\lambda}(k+1), \underline{u}(k), \underline{d}(k)) = \frac{1}{2}\left(\underline{x}^T(k)Q\underline{x}(k) + \underline{u}^T(k)R\underline{u}(k) - \gamma^2\underline{d}^T(k)\underline{d}(k)\right) + \underline{\lambda}^T(k+1)[\Phi\underline{x}(k) + \Gamma\underline{u}(k) + E\underline{d}(k)]$$

Optimality conditions:

$$\nabla_{\underline{\lambda}(k+1)}H = \underline{x}(k+1) = \Phi\underline{x}(k) + \Gamma\underline{u}(k) + E\underline{d}(k)$$

$$\nabla_{\underline{x}(k)}H = \underline{\lambda}(k) = Q\underline{x}(k) + \Phi^T\underline{\lambda}(k+1)$$

$$\nabla_{\underline{u}(k)}H = R\underline{u}(k) + \Gamma^T\underline{\lambda}(k+1) = \underline{0} \Rightarrow \underline{u}(k) = -R^{-1}\Gamma^T\underline{\lambda}(k+1)$$

$$\nabla_{\underline{d}(k)}H = -\gamma^2\underline{d}(k) + E^T\underline{\lambda}(k+1) = \underline{0} \Rightarrow \underline{d}(k) = \frac{1}{\gamma^2}E^T\underline{\lambda}(k+1)$$

$$\begin{bmatrix}\underline{x}(k+1)\\ \underline{\lambda}(k)\end{bmatrix} = \begin{bmatrix}\Phi & -S\\ Q & \Phi^T\end{bmatrix}\begin{bmatrix}\underline{x}(k)\\ \underline{\lambda}(k+1)\end{bmatrix}; S = (\Gamma R^{-1}\Gamma^T - \frac{EE^T}{\gamma^2})$$

$$\Rightarrow \begin{bmatrix}\underline{x}(k+1)\\ \underline{\lambda}(k+1)\end{bmatrix} = \underbrace{\begin{bmatrix}\Phi + S(\Phi^T)^{-1}Q & -S(\Phi^T)^{-1}\\ -(\Phi^T)^{-1}Q & (\Phi^T)^{-1}\end{bmatrix}}_{\text{Hamiltonian is a Symplectic matrix}}\begin{bmatrix}\underline{x}(k)\\ \underline{\lambda}(k)\end{bmatrix}$$

If we let $\underline{\lambda}(k) = P^*\underline{x}(k)$,

$$P^*\underline{x}(k) = Q\underline{x}(k) + \Phi^T P^*\underline{x}(k+1)$$

$$\underline{x}(k+1) = \Phi\underline{x}(k) - SP^*\underline{x}(k+1) \Rightarrow \underline{x}(k+1) = (I_n + SP^*)^{-1}\Phi\underline{x}(k)$$

$$\Rightarrow P^* = Q + \Phi^T P^*(I_n + SP^*)^{-1}\Phi$$

Computing Minimax Controller

- Main algorithm to find minimax controller gains

  Step 1: Pick a value of $\gamma > 0$ and compute the eigen values of the Hamiltonian

  Step 2: Check if Hamiltonian has any eigen values on the unit circle.
  If it does, increase γ and go to Step 1 with this γ. Else, go to Step 3.

  Step 3: Solve the discrete Riccati equation for P^*. Do Cholesky decomposition of P^*.
  If it is not positive definite, increase γ and go to Step 1. Else go to Step 4.

  Step 4: Check if $\Phi - \Gamma K$ is stable. If it is not, increase γ and go to Step 1. Else, we have found a minimax controller.

- Application to F-8 Example with $Q_1 = I_5$ and $R_1 = 0.01\, I_2$ in the *continuous* domain.
  - Discretize the system with h=0.01$\Rightarrow Q = \frac{h}{2}[\Phi' Q_1 \Phi + Q_1]; M = \frac{h}{2}\Phi' Q_1 \Gamma; R = hR_1 + \frac{h}{2}\Gamma' Q_1 \Gamma$
  - Form the Hamiltonian matrix. I found starting with a large value of γ better. DARE routine tells you when it can't order eigen values when they are close to unit circle
  - I found $\gamma = 0.165$ found the gains, but 0.160 didn't. Then, via bisection, you can find the smallest γ for which you can get stable controller is **0.1635**. This corresponds *to full state feedback H_∞ controller.* For γ greater than this minimum, it is a minimax controller.
  - Gain matrix (This controller will have a bias due to disturbances. Need integral control)

K=[-6.0591 -1.7236 -4.2557 3.3119 -1.2936
-1.9994 8.1329 -0.5474 4.9811 -0.3053]

Closed-loop Eigen values: [0.1813 0.9912 - 0.004i 0.9912 + 0.004i 0.9252 0.9927]