LQR (Linear Quadratic Regulator)

June Kwon
12/2/2021

- Optimal Control Theory

Given a system...

$$\dot{x} = f(x, u, t)$$

$$x(t_0) = x_0$$

* Note...

(where f, l, m are Lipschitz-continuous function

⇓ What do we mean by "Lipschitz"

- Continuous
- Differentiable
- Derivative value is (upper) bounded)

- Final Goal

→ Find optimal control input

$$u^*(t) \qquad (t_0 \le t \le T)$$

Such that it

- Minimize the cost (index) function

$$V(x(t_0), u, t_0) = \int_{t_0}^{T} L[x(\tau), u(\tau), \tau]\, d\tau + m(x(T))$$

- Hamilton-Jacobi Equation

$$V^*(x(t), t) \equiv \min_{u[t,T]} \left[V(x(t), u(t), t) \right]$$

"Let's say there exists cost function that has been minimized by certain control input"

- Principle of Optimality : break down into multiple sub-optimal problem, and when merged, it will still be optimal

$$V^*(x(t), t) = \min_{u[t,T]} \left\{ \int_{t}^{T} L[x(\tau), u(\tau), \tau]\, d\tau + m(x(T)) \right\}$$

• using "$t_0 < t < t_1 < T$", we can break down the problem into multiple subproblems!

$$V^*(x(t),t) = \min_{u[t,T]} \left\{ \int_t^{t_1} L[x(\tau),u(\tau),\tau]\,d\tau + \min_{u[t_1,T]} \left[\int_{t_1}^{T} L[x(\tau), u(\tau),\tau] + m(x(T)) \right] \right\}$$

$$= \min_{u[t,t_1]} \left\{ \int_t^{t_1} L[x(\tau),u(\tau),\tau]\,d\tau + V^*(x(t_1),t_1) \right\}$$

* set $\boxed{t_1 = t + \Delta t}$ very small value

$$V^*(x(t),t) = \min_{u[t,t+\Delta t]} \left\{ \int_t^{t+\Delta t} L[x(\tau),u(\tau),\tau]\,d\tau + V^*(x(t+\Delta t),t+\Delta t) \right\}$$

$$= \min_{u[t,t+\Delta t]} \left\{ \underbrace{\Delta t}_{\text{Base}} \cdot \underbrace{L[x(t+a\Delta t),u(t+a\Delta t),t+a\Delta t]}_{\text{Height}} + \underbrace{V^*(x(t+\Delta t),t+\Delta t)}_{\text{Taylor's Series}} \right\}$$

$$= \min_{u[t,t+\Delta t]} \Big\{ \Delta t \cdot L[x(t+a\Delta t),u(t+a\Delta t),t+a\Delta t] + V^*(x(t),t)$$

$$+ \left[\frac{\partial V^*}{\partial x}\right]^T (x(t),t)\,\Delta t \underbrace{\frac{dx(t)}{dt}}_{\text{due to chain rule.}} + \frac{\partial V^*}{\partial t}(x(t),t)\,\Delta t + H.O.T \Big\}$$

$\boxed{\tau = t + \alpha\Delta t}$

$0 < \alpha < 1$

- Thus,

$$\cancel{V^*(x(t),t)} = \min_{u[t,t+\Delta t]} \Big\{ \Delta t \cdot L[x(t+\alpha\Delta t), u(t+\alpha\Delta t), t+\alpha\Delta t]$$

$$+ \cancel{V^*(x(t),t)} + \left[\frac{\partial V^*}{\partial x}\right]^T (x(t),t)\Delta t \frac{dx(t)}{dt} + \boxed{\frac{\partial V^*}{\partial t}(x(t),t)\Delta t}$$

$$+ H.O.T \Big\}$$

- From above, devide both sides by Δt, and since Δt is infinitesimal, taking the limit $_{\Delta t \to 0}$ ...

$f(x,u,t) = \frac{dx(t)}{dt} = \dot{x}$

$$\left[\frac{\partial V^*}{\partial t}(x(t),t) = -\min_{u(t)} \left\{ \boxed{L[x(t),u(t),t] + \left[\frac{\partial V^*}{\partial x}\right]^T f(x,u,t)} \right\} \right]$$

"First Form of Hamiltonian Jacobi Equation"

- For $\boxed{\sim}$, partial differentiate the expression $(\sim)$ with respect to u, and assume u^* that makes $\frac{d}{du} = 0$ is the optimal input

$$u^*(t) \equiv \bar{u}\left(x(t), \frac{\partial V^*}{\partial x}, t\right)$$

- Substituting above $u^*(t)$ into above $[\sim]$ equation,

$$\frac{\partial V^*}{\partial t}(x(t),t)\Bigg|_{u^*(t)\equiv\bar{u}\left(x(t),\frac{\partial V^*}{\partial x},t\right)} = -\min_{u(t)}\left\{L[x(t),u(t),t]+\left[\frac{\partial V^*}{\partial x}\right]^T f(x,u,t)\right\}$$

- We then obtain below partial differential equation

$$\left[\begin{aligned} &\frac{\partial V^*}{\partial t}(x(t),t) = -L[x(t),\overset{\text{Optimal Input}}{\bar{u}(t)},t]+\left[\frac{\partial V^*}{\partial x}\right]^T f(x,\bar{u},t) \\ &\text{where} \quad V^*(x(T),T) = m(x(T)) \leftarrow \text{initial value}\end{aligned}\right]$$

After $u^*(t)$ is substitute

"cost functio
that's been
optimized"

- Then we solve above partial differential equation to find the optimal control input, $u^*(t)$.

→ Please go to "Example"!!!

* Example

→ Find optimal control input for regulation

$$\dot{x} = x + u \qquad (x^*(0) = 0)$$

→ Cost Function

$$V = \int_0^T (u^2 + x^2)\, d\tau \qquad (\text{* Assume } m(x(T)) = 0) \xrightarrow{\text{Thus}} L(x,u,t) = u^2 + x^2$$

→ Solution

$$\frac{\partial V^*}{\partial t}(x(t),t) = -\min_{u(t)} \left\{ L\left[x(t), u(t), t\right] + \left[\frac{\partial V^*}{\partial x}\right]^T \underbrace{f(x,u,t)}_{=\dot{x}} \right\}$$

↓

$$= -\min_{u(t)} \left\{ u^2 + x^2 + \left[\frac{\partial V^*}{\partial x}\right]^T \cdot (x+u) \right\}$$

* $\min_{u(t)} \{\}$: differentiate $\{\}$ w.r.t u and set $=0$ and solve!

$$0 = 2u + \left[\frac{\partial V^*}{\partial x}\right]^T \leftarrow \frac{d}{du}\{\} = 0$$

rearrange for u

$$\bar{u} = -\frac{1}{2}\frac{\partial V^*}{\partial x} = f\left(x(t), \frac{\partial V^*}{\partial x}, t\right)$$

→ Thus, plugging in $\bar{u}$ back into $\partial V/\partial t$. Since $\bar{u}$ is the optimal input that minizes $\partial V/\partial t$, min() will be omitted expression

$$\frac{\partial V^*}{\partial t}(x(t)\,t) = -\left[\frac{1}{4}\left(\frac{\partial V^*}{\partial x}\right)^2 + x^2 + \left[\frac{\partial V^*}{\partial x}\right]^T \cdot \left(x - \frac{1}{2}\frac{\partial V^*}{\partial x}\right)\right]$$

$$= -\min_{u(t)} \left\{ \underbrace{u^2 + x^2}_{L} + \left[\frac{\partial V^*}{\partial x}\right]^T \cdot \underbrace{(x+u)}_{f(x,u,t)} \right\}$$

$$= -\left\{ \bar{u}^2 + x^2 + \left[\frac{\partial V^*}{\partial x}\right]^T \cdot (x+\bar{u}) \right\}$$

→ Then solve above partial differential equation...

$V^*(x(t),t) = ???$ "What would be the optimal cost function?"

Very hard to solve. Thus use adaptive, NN, etc...

→ Or we can approximate it for use in linear case by setting

$$\frac{\partial V^*}{\partial t}(x(t),t) = -\left\{ \bar{u}^2 + x^2 + \left[\frac{\partial V^*}{\partial x}\right]^T (x+\bar{u}) \right\} = 0 \Rightarrow \text{Solve for } \bar{u} \text{ that satisfies this eq}$$

we will have to assume V^*

LQR - Apply Hamilton-Jacobi EQ to linear case.

- Given system,

$$\dot{x} = Ax + Bu \qquad x(t_0) = x_0$$

- Final Goal

→ Find optimal control input: $u^*(t)$, $t_0 \le t \le T_f$

→ Minimize the cost (index) function:

$$V(x(t_0), u, t_0) = \int_{t_0}^{T_f} (u^T R u + x^T Q x) dt + x^T(T_f) Q x(T_f)$$

→ Let's assume optimal cost function exists as shown below

$$\left[V^*(x(t_0), u, t_0) = x^T(t) P x(t) \right] \qquad ({}^* P = P^T)$$

that there is P which satisfies Hamilton-Jacobi equation..

Very important assumption root of LQR

Thus, we bring from above...

- LQR : Hamilton-Jacobi EQ..

$$\frac{\partial V^*}{\partial t}(x(t), t) = -\min_{u(t)} \left\{ L[x(t), u(t), t] + \left[\frac{\partial V^*}{\partial x}\right]^T f(x, u, t) \right\}$$

$$\begin{cases} V(x(t_0), u, t_0) = \int_{t_0}^{T_f} \underbrace{(u^T R u + x^T Q x)}_{L} dt + x^T(T_f) Q x(T_f) \\ \dot{x} = \underbrace{Ax + Bu}_{f}, \quad x(t_0) = x_0 \end{cases}$$

Thus,

$$\begin{cases} L = u^T R u + x^T Q x \qquad f = Ax + Bu \\ V^* = x^T P x \end{cases}$$

→ Let's substitute above $^{(L, f, V^*)}$ to find the optimal input, $\bar{u}$ which satisfies Hamilton-Jacobi equation..

$$\frac{\partial V^*}{\partial t}(x(t), t) = -\min_{u(t)} \left\{ \underbrace{u^T R u + x^T Q x}_{L[x(t), u(t), t]} + \underbrace{2x^T P}_{[\partial V^*/\partial x]} \underbrace{(Ax + Bu)}_{f(x, u, t)} \right\}$$

→ Substitute $\bar{u}$ to u so that we imply $\bar{u}$ yields minimum (now we can take out min{}...)

$$\frac{\partial V^*}{\partial t} = -\{\bar{u}^T R \bar{u} + x^T Q x + 2x^T P(Ax + B\bar{u})\}$$

→ In order to find $\bar{u}$ which minimizes $\partial V^* / \partial t$, we set equal to 0 and solve for $\bar{u}$

factor -out..

$$\bar{u}^T R \bar{u} + 2x^T P B \bar{u} + x^T Q x + 2x^T P A x = 0$$

Factor out..
(ex: $x^2 + 4x + 15 = \overbrace{x^2 + 4x + 4}^{(x+2)^2} - 4 + 15$)

$$\underbrace{u^T R u + 2x^T P B u + x^T P B R^{-1} B^T P x}_{\boxed{(u + R^{-1} B^T P x)^T R (u + R^{-1} B^T P x)}} - x^T P B R^{-1} B^T P x + x^T Q x + 2x^T P A x = 0$$

→ Thus... factoring above eq, we have..

$$0 = \left[(\bar{u} + R^{-1} B^T P x)^T R (\bar{u} + R^{-1} B^T P x) + x^T (Q + PA + A^T P - P B R^{-1} B^T P) x \right]$$

Thus, the control input $\bar{u}$ which satisfies above equation and minimizes $\partial V / \partial t$ is as follows..

such that $u^* = -Kx$ like a state feedback!

$$\left[\bar{u} = -\overbrace{R^{-1} B^T P}^{K} x \right]$$

However.. ↓

→ However, for above eq to be zero (→ To be minimized...), following condition also needs to hold...

Given: A, B, Q, R Find: P

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

This is called "Algebraic Riccati Equation" (ARE)

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}$$

using Hamiltonian Matrix, we can solve ARE. That is, we want to find "P" such that ARE = 0 is satisfied

→ Thus, for LQR control law, $\bar{u} = -Kx = -R^{-1}B^TPx$
we want to find P such that

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

"ARE" is satisfied

→ If (A, B, C) are "controllable" and "observable"

and $Q \geq 0$ (positive semidef) $R > 0$ (positive def) $\int_0^\infty x^TQx + u^TRu\,dt$ infinite horizon

↳ Thus, no terminal constraint

Then there exists unique positive definite solution P $(P > 0)$ that satisfies ARE and also stabilizes the closed-loop system $(\underbrace{A - BR^{-1}B^TP}_{})$

$\dot{x} = (A - BK)x = Ax + Bu$

→ Use Hamiltonian Matrix to solve ARE

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}$$

→ Take eigenvalue & vector

must check!

$$\underset{H}{\begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}} \underset{\text{E.Vector}}{\begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}} = \underset{\text{E.Vector}}{\begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}} \underset{\text{E.Value}}{\begin{bmatrix} \lambda_{stable} & 0 \\ 0 & \lambda_{unstable} \end{bmatrix}}$$

→ Then there exists unique positive definite stabilizing P,

If V_{11}^{-1} exists,

$$\boxed{P = V_{21}V_{11}^{-1}} \text{ solves ARE}$$

→ Proof

Consider stable eigenvalue portion with its corresponding eigenvector...

$$\begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}\begin{bmatrix} V_{11} \\ V_{21} \end{bmatrix} = \begin{bmatrix} V_{11} \\ V_{21} \end{bmatrix}\lambda_s$$

$$AV_{11} - BR^{-1}B^TV_{21} = V_{11}\lambda_s$$

$\times V_{11}^{-1}$

$$A - BR^{-1}B^TV_{21}V_{11}^{-1} = V_{11}\lambda_s V_{11}^{-1}$$

$$-QV_{11} - A^TV_{21} = V_{21}\lambda_s$$

$\times V_{11}^{-1}$

$$-Q - A^TV_{21}V_{11}^{-1} = V_{21}\lambda_s V_{11}^{-1}$$

$\times V_{21}V_{11}^{-1}$

$$V_{21}V_{11}^{-1}A - V_{21}V_{11}^{-1}BR^{-1}B^TV_{21}V_{11}^{-1} = V_{21}\lambda_s V_{11}^{-1}$$

same!

Thus,

$$V_{21}V_{11}^{-1}A - V_{21}V_{11}^{-1}BR^{-1}B^TV_{21}V_{11}^{-1} = -Q - A^TV_{21}V_{11}^{-1}$$

$\updownarrow$ compare with ARE

$$\text{ARE} \rightarrow \quad PA - PBR^{-1}B^TP = -Q - A^TP$$

Therefore, $\boxed{P = V_{21} V_{11}^{-1}}$ ✓

* In case λ_{stable} and $\lambda_{unstable}$ are flipped over..

$$\begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix} = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix} \begin{bmatrix} \lambda_{unstable} & 0 \\ 0 & \lambda_{stable} \end{bmatrix}$$

Then. $\boxed{P = U_{22} U_{12}^{-1}}$ // $\therefore U_{12}^{-1}$ must exist.

→ Thus, with $P = V_{21} V_{11}^{-1}$, the poles of the closed-loop system will be... $\dot{x} = Ax + Bu = (A - BR^{-1}B^T P)x$

Ans: the stable part of the eigenvalues of the hamiltonian matrix!

$$\begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} V_{11} \\ V_{21} \end{bmatrix} = \begin{bmatrix} V_{11} \\ V_{21} \end{bmatrix} \lambda_{stable}$$

$$AV_{11} - BR^{-1}B^T V_{21} = V_{11} \lambda_{stable}$$

$\times V_{11}^{-1}$

$$A - BR^{-1}B^T V_{21} V_{11}^{-1} = V_{11} \lambda_{stable} V_{11}^{-1}$$

$$A - BR^{-1}B^T P = \underline{V_{11} \lambda_{stable} V_{11}^{-1}}$$

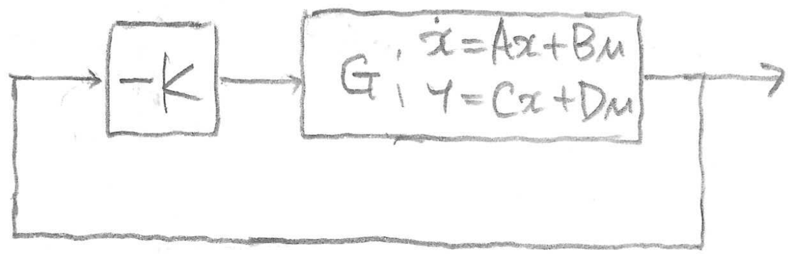
→ This is essentially similarity transformation

Thus, $\boxed{eig(A - BR^{-1}B^T P) = eig(\lambda_{stable})}$

→ Thus, once P has been found, LQR control law can be found..

$$\left[\begin{aligned} u &= -R^{-1}B^{T}Px \\ &= -\underset{\uparrow \text{LQR Gain}}{K}x \end{aligned}\right] \text{LQR control law} \qquad \overset{\text{where}}{(K = R^{-1}B^{T}P)}$$

→ Thus



Closed-loop system is stable..

$$\dot{x} = Ax + Bu = Ax + B(-R^{-1}B^{T}P)x = \underbrace{(A - BR^{-1}B^{T}P)}_{\text{eig} = \text{stable}}x$$

$$y = \underset{\uparrow\,\text{Choice}}{C}x + \underset{\uparrow\,[\,]}{D}u$$

→ Finally.. Procedure for LQR

1. Given A & B from plant
2. Choose Q & R
3. Solve ARE for P (which stabilizes closed-loop)
4. Compute Gain $K = R^{-1}B^{T}P$ s.t. $u = -Kx$

→ Question: Why do we choose such cost function?
Ans: Energy

$$V(x(t_0), u(t_0)) = \int_{t_0}^{T_f} (\underbrace{u^T R u}_{} + \underbrace{x^T Q x}_{}) dt + \underbrace{x^T(T_f) Q x(T_f)}_{}$$

→ Here, $u^T R u$ and $x^T Q x$ are actually form of energy.

→ For example,

$$P = VI = I^2 R = \frac{V^2}{R}$$

→ When you look at it, power (P) is expressed in terms of square of current (I^2) or voltage (V^2). Here, resistance (R) is a weighting parameter that we can choose.

→ Thus, in LQR, R and Q are like resistance. They are weighting parameter that we can choose (and thus, the quadratic state variables ($x^T x$, $u^T u$) represent <u>energy</u> of that state variable.

→ Thus, in LQR, we control Q and R (which should be more considered?) and solve the optimization problem... (minimization of cost function)

→ Terms in Cost Function? What do they mean?

$$J = \int_0^{\infty} x^T Q x + u^T R u \, dt$$

where

x: $n \times 1$ state vectors $n\left[\;\right]_1$

u: $m \times 1$ control vectors $m\left[\;\right]_1$

Q: $n \times n$ symmetric positive semi definite matrix

$(Q \geq 0,\ Q \succeq 0)$ $n\left[\begin{smallmatrix} * & & \\ & * & \\ & & * \end{smallmatrix}\right]_n$

↳ Positive semidefinite:

$$\underset{1\times n}{x^T} \underset{n\times n}{Q} \underset{n\times 1}{x} \geq 0 \quad \forall x$$

the scalar, $x^T Q x$, (1×1) is always greater (≥ 0) than or equal to 0 for all possible x $(\forall x)$

R: $m \times m$ symmetric positive definite matrix

$(R > 0,\ R \succ 0)$

↑ notice there is no "equal to" sign

↳ Positive Definite :

$$\underset{1\times m}{u^T}\,\underset{m\times m}{Q}\,\underset{m\times 1}{u} > 0 \quad \forall u$$

the scalar, $u^T Q u$, (1×1) is $\underset{\text{(no equal to)}}{\text{strictly}}$ greater than $\underset{}{0}$ (>0)
for all possible u $(\forall u)$

→ Thus,

$$Q = \begin{bmatrix} q_{11} & & & \\ & q_{22} & & \\ & & \ddots & \\ & & & q_{nn} \end{bmatrix} n \qquad R = \begin{bmatrix} r_{11} & & & \\ & r_{22} & & \\ & & \ddots & \\ & & & r_{mm} \end{bmatrix} m$$

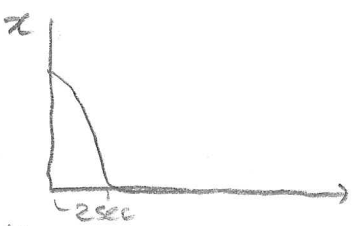
$$\quad n \qquad\qquad\qquad\qquad\qquad m$$

"As long as entries in Q and R $(q_{11}, q_{22}, \ldots q_{nn}$ and, r_{11}, $r_{22}, \ldots r_{mm})$ are positive, "Q will be positive semidefinite", and "R will be positive definite""

→ Also, for $J = \underbrace{\int_0^\infty x^T Q x}_{} + u^T R u \, dt$ and minimize J

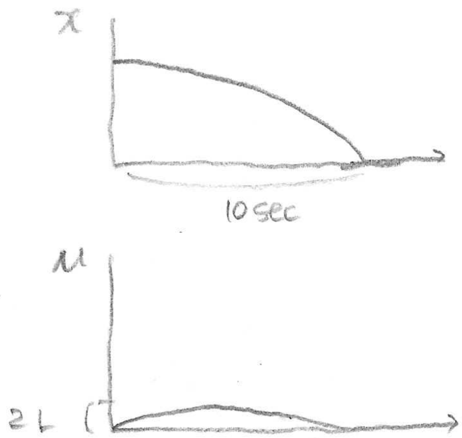
if we have

$Q > R$: This will dominate in J, and whenever there is nonzero x, $\min\{J\}$ will regulate (remove) the non-zero. Thus,



- Faster regulation of $x \to 0$
- u will be large
- Aggressive
- "I want x to be 0 as soon as possible, no matter the cost"

If $Q < R$: u^TRu term will dominate in J, and whenever there is non-zero u, $\min\{J\}$ will regulate (remove) the non-zero. Thus



- Slower regulation of $x \to 0$
- u will be small
- conservative
- "I want to minimize fuel consumption as much as possible no matter the performance"

→ How to set up Q & R matrices

For given 2 states and 2 control inputs system..

$$\underset{n\times 1}{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \begin{matrix} \text{position} \\ \text{velocity} \end{matrix} \qquad \underset{m\times 1}{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \begin{matrix} \text{input 1} \\ \text{input 2} \end{matrix}$$

For Q & R

$$\underset{n\times n}{Q} = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix} \qquad \underset{m\times m}{R} = \begin{bmatrix} r_{11} & 0 \\ 0 & r_{22} \end{bmatrix}$$

→ Thus, for cost function..

$$x^TQx + u^TQu = q_{11}x_1^2 + q_{22}x_2^2 + r_{11}u_1^2 + r_{22}u_2^2$$

q_{11}: weight/penalty on non-zero x_1

q_{22}: weight/penalty on non-zero x_2

Thus, if $q_{11} > q_{22}$

→ I want $\underset{\text{position}}{x_1}$ to be zero asap no matter $\underset{\text{velocity}}{x_2}$

if $q_{11} < q_{22}$

→ I want $\underset{\text{velocity}}{x_2}$ to be zero asap no matter $\underset{\text{position}}{x_1}$

r_{11} : weight/penalty on non-zero u_1

r_{22} : weight/penalty on non-zero u_2

Thus, if $r_{11} > r_{22}$

→ I want u_1 to be zero asap no matter u_2

if $r_{11} < r_{22}$

→ I want u_2 to be zero asap no matter u_1

Understanding LQR - Simple Example : How does different Q & R affect the system performance?

```
% June Kwon

clc
clear
close all
%#ok<*ASGLU>
```

1. Given System Dynamics (Mass-Damper System)

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -0.2 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

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

```
% x1 : Position (m)
% x2 : Velocity (m/s)
A = [0 1 ; 0 -1/5];
B = [0 1]';
C = eye(2);
D = [];
G = ss(A,B,C,D)
```

```
G =

  A =
         x1    x2
   x1     0     1
   x2     0  -0.2

  B =
       u1
   x1   0
   x2   1

  C =
       x1  x2
   y1   1   0
   y2   0   1

  D =
       u1
   y1   0
   y2   0

Continuous-time state-space model.
```

2. Define Q & R Weighting Matrices

```
% (Randomly Chosen Q & R)
```

```
Q = eye(2)   ,...
R = 1/100
```

```
Q = 2×2
     1     0
     0     1
R = 0.0100
```

3. Solve Algebraic Riccati Equation for P using Hamiltonian Matrix

Algebraic Riccati Equation :

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

Hamiltonian Matrix :

$$H = \begin{bmatrix} A & -\mathrm{BR}^{-1} B^T \\ -Q & -A^T \end{bmatrix}$$

```
% Define Hamiltonian Matrix
H = [A -B/R*B' ; -Q -A']
```

```
H = 4×4
         0    1.0000         0         0
         0   -0.2000         0 -100.0000
   -1.0000         0         0         0
         0   -1.0000   -1.0000    0.2000
```

```
% Eigen-decompose Hamiltonian Matrix
[HV,HE] = eig(H)
```

```
HV = 4×4
    0.0995   -0.5773    0.5773   -0.0995
    0.9899   -0.5801   -0.5802    0.9903
   -0.0100    0.5745    0.5745   -0.0100
   -0.1005    0.0070   -0.0047    0.0966
HE = 4×4
    9.9514         0         0         0
         0    1.0049         0         0
         0         0   -1.0049         0
         0         0         0   -9.9514
```

```
% Stable Eigenvalues are located in right bottom.
% Thus P = U22 / U12
U12 = HV(1:2,3:4);
U22 = HV(3:4,3:4);
% Then, P that satisfies ARE can be found
P = U22 / U12
```

```
P = 2×2
    1.0956    0.1000
    0.1000    0.1076
```

```
% [Check] Is P a positive definite (P > 0)?
```

```
% 1. Perfrom Cholesky Factorization
%   >> If Cholesky Factorizable, Positive Definite
%   >> If not Cholesky Factorizable, Not a Positive Definite
try chol(P)
    disp('Matrix is symmetric positive definite.')
catch ME
    disp('Matrix is not symmetric positive definite')
end
```

```
ans = 2×2
    1.0467    0.0955
         0    0.3137
Matrix is symmetric positive definite.
```

```
% [Check] Is P a positive semi-definite (P >= 0)?
% 1. Perfrom Eigenvalue Decomposition
%   >> If All Eigenvalues are greater than 0, Positive Definite
%   >> If Some Eigenvalues are equal to 0, Positive Semi-Definite
if issymmetric(P)
    EV_P = eig(P)
    if all(EV_P > 0)
        disp('All Eignevalues are greater than 0, Thus, P is a positive definite.')
    else
        disp('Some Eignevalues are equal to 0, Thus, P is a positive semi-definite.')
    end
end
```

```
EV_P = 2×1
    0.0975
    1.1056
All Eignevalues are greater than 0, Thus, P is a positive definite.
```

```
% Finally, check Does P satisfy ARE?
ARE = Q + P*A + A'*P - P*B/R*B'*P
```

ARE = 2×2

10^{-14} ×

```
   -0.1110   -0.0444
   -0.0444   -0.0222
```

Yes, with the computed P above, the ARE equals to all zero.

4. Compute LQR Gain & Closed-Loop System Stability

$$K = R^{-1}B^TP$$

$$\dot{x} = \mathrm{Ax} + \mathrm{Bu} = (A - \mathrm{BK})x$$

```
% LQR Gain
K = R\B'*P
```

```
K = 1×2
   10.0000   10.7563
```

```
% Closed-loop System with LQR Gain
```

```
GCL = ss(A-B*K,[],eye(2),[])
```

```
GCL =

  A =
            x1      x2
   x1        0       1
   x2      -10  -10.96

  B =
     Empty matrix: 2-by-0

  C =
        x1  x2
   y1    1   0
   y2    0   1

  D =
     Empty matrix: 2-by-0

Continuous-time state-space model.
```

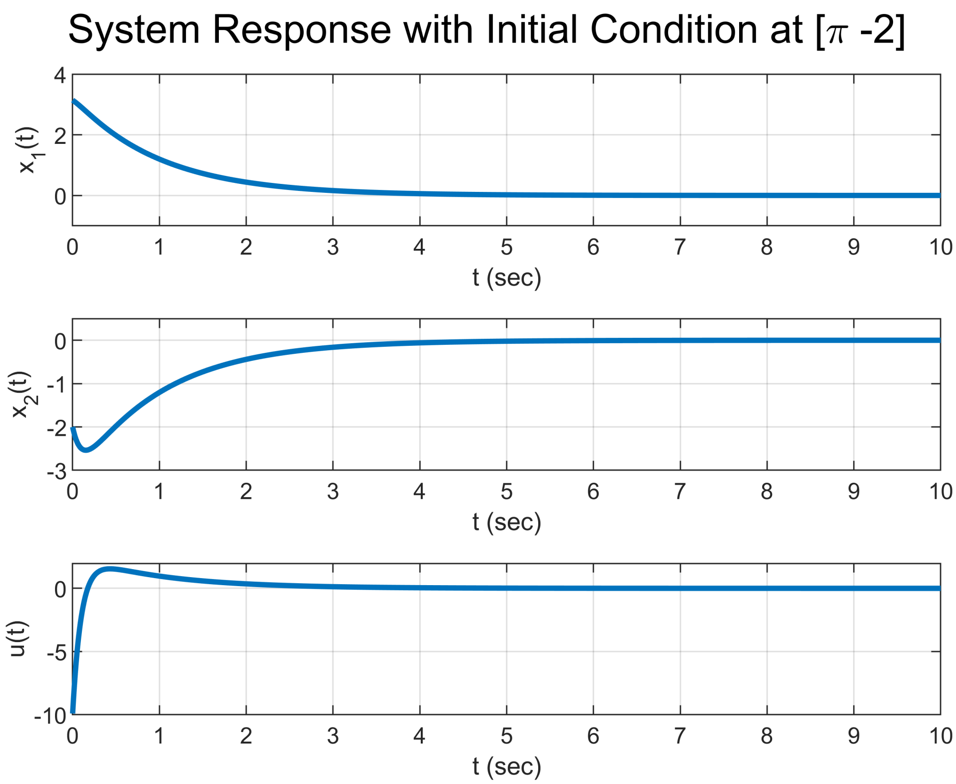
```
% Eigenvalues of Closed-Loop System to evaulate the stability of the system
E = eig(GCL)
```

```
E = 2×1
   -1.0049
   -9.9514
```

All Negative, Thus, Closed-loop System will be stable.

5. Simulation with Initial Condition at $[\pi \quad -2]^T$

```
T = 0:0.01:10;
[y,t,x] = initial(GCL,[pi -2]',T);
u = (-K*x')';
figure;
subplot(3,1,1);
plot(t,y(:,1),'linewidth',2);
ylabel('x_1(t)'); xlabel('t (sec)');
grid on; ylim([-1 4]);
subplot(3,1,2);
plot(t,y(:,2),'linewidth',2)
ylabel('x_2(t)'); xlabel('t (sec)');
grid on; ylim([-3 0.5]);
subplot(3,1,3);
plot(t,u,'linewidth',2)
ylabel('u(t)'); xlabel('t (sec)');
grid on; ylim([-10 2]);
sgtitle('System Response with Initial Condition at [\pi -2]');
```



As shown above, given the initial condition, the system successfully regulates the initial condition down to zero.

6. Simulation with Varying Q & R

```
% Time Range
T = 0:0.01:20;
```

6 - 1. [$Q > R$] Penalize Q : "I want the states (x) to be regulated as soon as possible, no matter how much it costs (u)"

```
Q1 = eye(2), R1 = 1/100, [K1,P1,E1] = lqr(G,Q1,R1)
```

```
Q1 = 2×2
     1     0
     0     1
R1 = 0.0100
K1 = 1×2
   10.0000   10.7563
P1 = 2×2
    1.0956    0.1000
    0.1000    0.1076
E1 = 2×1
   -1.0049
   -9.9514
```

```
GCL1 = ss(A-B*K1,[],eye(2),[]);
[y1,t1,x1] = initial(GCL1,[pi -2]',T);
```

```
u1 = (-K1*x1')';
```

6 - 2. [$Q < R$] Penalize R : "I want the least amount of cost (u) to be spent, no matter how the system (x) performs"

```
Q2 = eye(2), R2 = 100, [K2,P2,E2] = lqr(G,Q2,R2)
```

```
Q2 = 2×2
     1     0
     0     1
R2 = 100
K2 = 1×2
    0.1000    0.3000
P2 = 2×2
    5.0000   10.0000
   10.0000   30.0000
E2 = 2×1 complex
  -0.2500 + 0.1936i
  -0.2500 - 0.1936i
```

```
GCL2 = ss(A-B*K2,[],eye(2),[]);
[y2,t2,x2] = initial(GCL2,[pi -2]',T);
u2 = (-K2*x2')';
```

6 - 3. [$q_{11} > q_{22}$] Penalize q_{11} : "I want the position (x_1) to be regulated as soon as possible, no matter what the velocity (x_2) is"

```
Q3 = [10 0; 0 0.1], R3 = 1, [K3,P3,E3] = lqr(G,Q3,R3)
```

```
Q3 = 2×2
   10.0000         0
         0    0.1000
R3 = 1
K3 = 1×2
    3.1623    2.3425
P3 = 2×2
    8.0402    3.1623
    3.1623    2.3425
E3 = 2×1 complex
  -1.2713 + 1.2434i
  -1.2713 - 1.2434i
```

```
GCL3 = ss(A-B*K3,[],eye(2),[]);
[y3,t3,x3] = initial(GCL3,[pi -2]',T);
u3 = (-K3*x3')';
```

6 - 4. [$q_{11} < q_{22}$] Penalize q_{22} : "I want the velocity (x_2) to be regulated as soon as possible, no matter what the position (x_1) is"

```
Q4 = [0.1 0; 0 10], R4 = 1, [K4,P4,E4] = lqr(G,Q4,R4)
```

```
Q4 = 2×2
    0.1000         0
         0   10.0000
```

```
R4 = 1
K4 = 1×2
    0.3162    3.0669
P4 = 2×2
    1.0331    0.3162
    0.3162    3.0669
E4 = 2×1
   -0.0999
   -3.1670
```

```
GCL4 = ss(A-B*K4,[],eye(2),[]);
[y4,t4,x4] = initial(GCL4,[pi -2]',T);
u4 = (-K4*x4')';
```

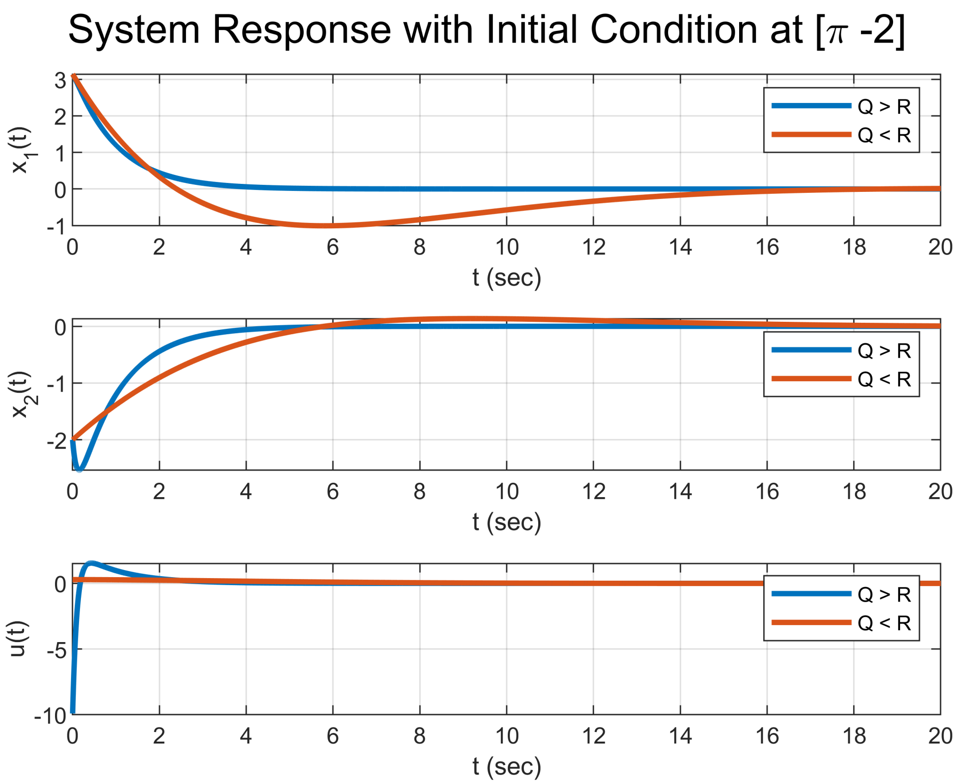
6 - 5. Plotting for Q vs R

```
figure;
subplot(3,1,1);
plot(t1,y1(:,1),'linewidth',2); hold on;
plot(t2,y2(:,1),'linewidth',2);
legend("Q > R","Q < R");
ylabel('x_1(t)'); xlabel('t (sec)'); grid on;

subplot(3,1,2);
plot(t1,y1(:,2),'linewidth',2); hold on;
plot(t2,y2(:,2),'linewidth',2);
legend("Q > R","Q < R");
ylabel('x_2(t)'); xlabel('t (sec)'); grid on;

subplot(3,1,3);
plot(t1,u1,'linewidth',2); hold on;
plot(t2,u2,'linewidth',2);
legend("Q > R","Q < R");
ylabel('u(t)'); xlabel('t (sec)'); grid on;

sgtitle('System Response with Initial Condition at [\pi -2]');
```



As shown above,

When $Q > R$, we expect the states (x) to be regulated as soon as possible, no matter how much the cost (u) is. As shown above, we can see that the states x_1 and x_2 were successfully regulated to zero faster than $Q < R$ case. However, this caused a high cost (u) to achieve such performance.

When $Q < R$, we expect the least amount of cost (u) to be spent, no matter how the system (x) performs. As shown above, compared to $Q > R$ case, we can see that the smaller amount of cost (u) was spent to regulate the states. However, this caused the system (x) to take longer time (slower response) to be regulated.

6 - 6. Plotting for q_{11} vs q_{22}

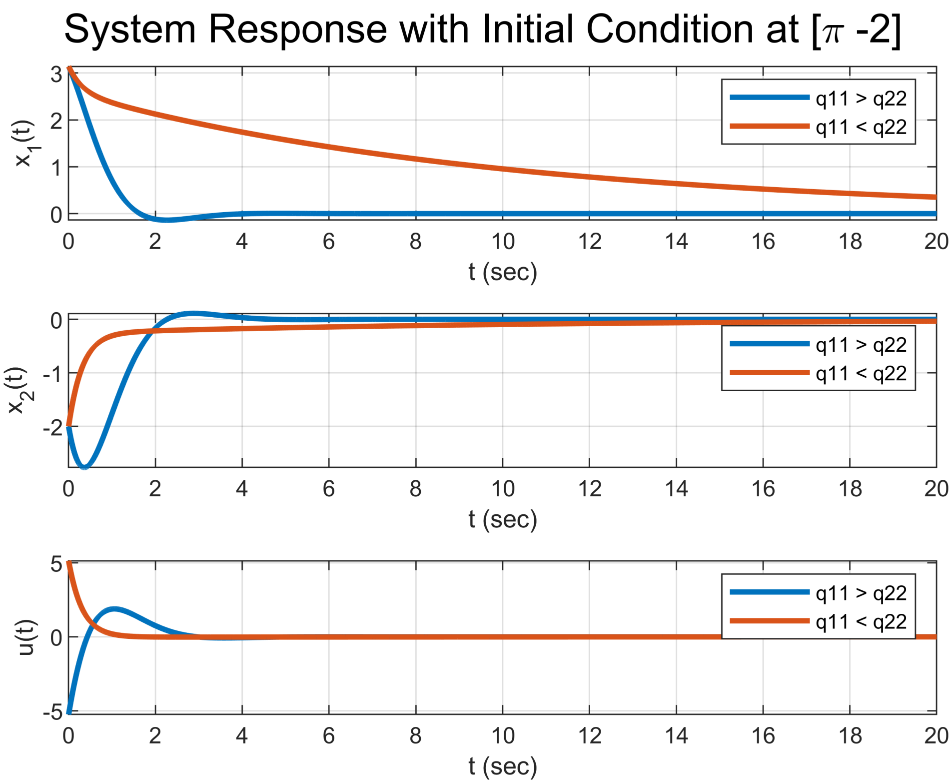
```
figure;
subplot(3,1,1);
plot(t3,y3(:,1),'linewidth',2); hold on;
plot(t4,y4(:,1),'linewidth',2);
legend("q11 > q22","q11 < q22");
ylabel('x_1(t)'); xlabel('t (sec)'); grid on;

subplot(3,1,2);
```

```
plot(t3,y3(:,2),'linewidth',2); hold on;
plot(t4,y4(:,2),'linewidth',2);
legend("q11 > q22","q11 < q22");
ylabel('x_2(t)'); xlabel('t (sec)'); grid on;

subplot(3,1,3);
plot(t3,u3,'linewidth',2); hold on;
plot(t4,u4,'linewidth',2);
legend("q11 > q22","q11 < q22");
ylabel('u(t)'); xlabel('t (sec)'); grid on;

sgtitle('System Response with Initial Condition at [\pi -2]');
```



As shown above,

When $q_{11} > q_{22}$, we expect the position (x_1) to be regulated faster than the velocity (x_2). As shown above, we can see that the position (x_1) is successfully regulated down to 0 faster. (Since the velocity is a derivative of the position, the velocity (x_2) here just depends on the position (x_1) and comes down to 0 as the position comes down to 0. Thus, it might be hard to explicitly say that the position (x_1) can be regulated faster than the velocity (x_2) for this example...)

When $q_{11} < q_{22}$, we expect the velocity (x_2) to be regulated faster than the position (x_1). As shown above, we can see that the velocity (x_2) is being regulated faster than the poisition (x_1). Thus, the position (x_1) is very slowly coming down to 0. (In other words, $q_{11} < q_{22}$ puts more weight (importance) on the velocity (x_2), thus, in this example, we want the velocity (x_2) to be as low as possible, and thus, with the slower velocity, the position (x_1) will take longer time to come down to 0 as shown above.

(7. Comparison with MATLAB's Built-in LQR Function & iCARE Function)

```
% [JUNE] P, K, E
P, K, E
```

```
P = 2×2
    1.0956    0.1000
    0.1000    0.1076
K = 1×2
   10.0000   10.7563
E = 2×1
   -1.0049
   -9.9514
```

```
% [MATLAB] ARE Function
[PM,KM,EM] = icare(A,B,Q,R)
```

```
PM = 2×2
    1.0956    0.1000
    0.1000    0.1076
KM = 1×2
   10.0000   10.7563
EM = 2×1
   -1.0049
   -9.9514
```

Given A,B,Q,R, MATLAB's built-in iCARE function yielded below three outputs.

1. PM : P Matrix that satisfies ARE

2. KM : K Gain using P $(K = R^{-1}B^TP)$

3. EM : Eigenvalues of Closed-Loop System $(\text{eig}(A - \text{BK}))$

Comparing above PM, KM, EM to the manually computed P,K,E, the manual computation of ARE was successful.

```
% [MATLAB] LQR Function
[KL,PL,EL] = lqr(G,Q,R)
```

```
KL = 1×2
   10.0000   10.7563
PL = 2×2
```

```
     1.0956    0.1000
     0.1000    0.1076
EL = 2×1
   -1.0049
   -9.9514
```

Given G,Q,R, MATLAB's built-in LQR function yielded below three outputs.

1. KL : K Gain using P $(K = R^{-1}B^TP)$

2. PL : P Matrix that satisfies ARE

3. EL : Eigenvalues of Closed-Loop System $(\mathrm{eig}(A - \mathrm{BK}))$

Comparing above KL, PL, EL to the manually computed K,P,E, the manual computation of LQR was successful.