LQR

AE353

Spring 2025

Bretl

$$\dot{x} = Ax + Bu$$
$$u = -Kx$$

HOW TO FIND K ?

1. Gain tuning (i.e., guess and check)
   - make a small change to K
   - check if all eigenvalues of A-BK have negative real part
   - repeat until satisfied

2. Eigenvalue placement
   - choose desired eigenvalue locations
   - apply "place_poles" or Ackermann's method

3. LQR (minimize a cost)
   - choose weights on the cost of non-zero x and u
   - choose K to minimize total, integrated cost

EXAMPLE (RUN DEMO)

$$\left.\begin{aligned} \dot{x} &= \overset{A}{[5]}\,x + \overset{B}{[1]}\,u \\ u &= -\underset{K}{[k]}\,x \end{aligned}\right\} \quad \dot{x} = \underbrace{\overset{A-BK}{[5-k]}}\,x$$

$k > 5$ for stability

$k = 5 - p$ for eigenvalue at p

Linear Quadratic Regulator (LQR)

$$\underset{u_{[t_0,\infty)}}{\text{minimize}} \quad \boxed{\int_{t_0}^{\infty} \boxed{\left(x(t)^T Q\, x(t) + u(t)^T R\, u(t) \right)} \, dt}$$

$$\text{subject to} \quad \boxed{\begin{aligned} \dot{x}(t) &= A x(t) + B u(t) \\ x(t_0) &= x_0 \end{aligned}}$$

cost

total cost

constraints

The minimizer (i.e., the input that achieves minimum cost) is

$$u(t) = -K\, x(t)$$

and the minimum (i.e., the minimum cost) is

$$x_0^T P x_0$$

where K and P can be found in python as follows:

```python
def lqr(A, B, Q, R):
    P = linalg.solve_continuous_are(A, B, Q, R)
    K = linalg.inv(R) @  B.T @ P
    return K, P
```

$$\underset{u_{[t_0,\infty)},\, x_{[t_0,\infty)}}{\text{minimize}} \quad \int_{t_0}^{\infty} \left(x(t)^T Q\, x(t) + u(t)^T R\, u(t) \right) dt$$

$$\text{subject to} \quad \dot{x}(t) = A\,x(t) + B\,u(t)$$

$$x(t_0) = x_0$$

Why is the cost "quadratic" and what does it really mean?

$$x = [x_1] \qquad Q = [q_1] \qquad u = [u_1] \qquad R = [r_1]$$

$$x^T Q\, x + u^T R\, u = [x_1][q_1][x_1] + [u_1][r_1][u_1]$$

$$= [q_1 x_1^2] + [r_1 u_1^2]$$

$$= [q_1 x_1^2 + r_1 u_1^2]$$

Why is the cost "quadratic" and what does it really mean?

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad u = [u_1]$$

$$Q = \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix} \qquad R = [r_1]$$

$$x^T Q x + u^T R u = [x_1 \quad x_2] \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [u_1][r_1][u_1]$$

$$= [x_1 \quad x_2] \begin{bmatrix} q_1 x_1 + q_3 x_2 \\ q_3 x_1 + q_2 x_2 \end{bmatrix} + [r_1 u_1^2]$$

$$= [q_1 x_1^2 + q_3 x_1 x_2 + q_3 x_1 x_2 + q_2 x_2^2] + [r_1 u_1^2]$$

$$= [q_1 x_1^2 + 2 q_3 x_1 x_2 + q_2 x_2^2 + r_1 u_1^2]$$

What Q and R would produce a given cost?

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad u = [u_1]$$

$$Q = \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix} \qquad R = [r_1]$$

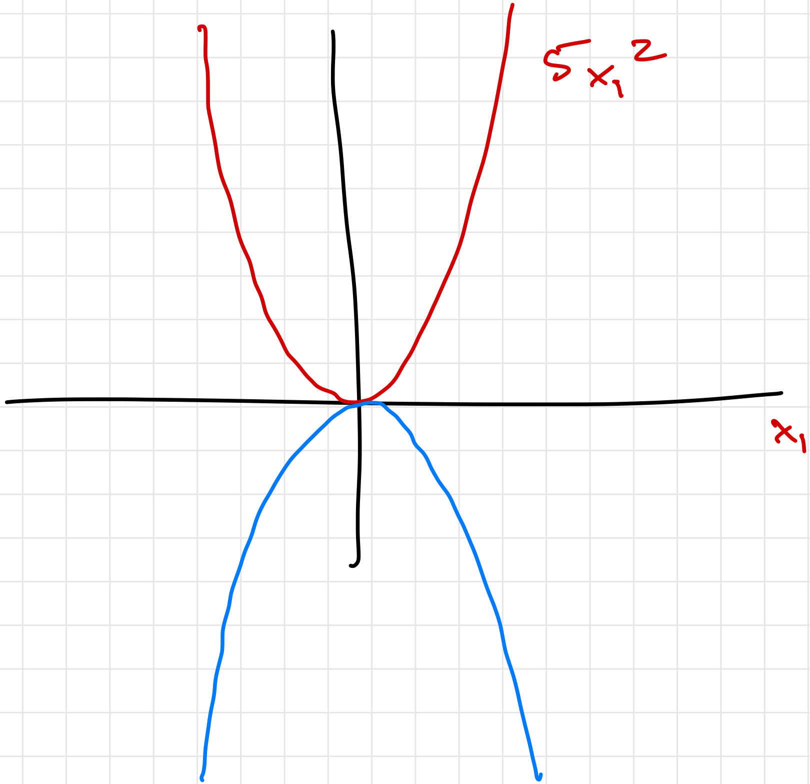
$$[q_1 x_1^2 + 2 q_3 x_1 x_2 + q_2 x_2^2 + r_1 u_1^2]$$

$$x^T Q x + u^T R u = 5 x_1^2 - 2 x_1 x_2 + 2 x_2^2 + 1 u_1^2$$

$$\downarrow$$

$$Q = \begin{bmatrix} 5 & -1 \\ -1 & 2 \end{bmatrix} \qquad R = [1]$$

$5x_1^2$



Q and R are commonly chosen to be diagonal

all positive

$$Q = \mathrm{diag}(q_1, \ldots, q_{n_x})$$

$$R = \mathrm{diag}(r_1, \ldots, r_{n_u})$$

all positive

Rule of thumb ("Bryson's rule")

$$q_i = \left(\frac{1}{x_{i,max}} \right)^2 \qquad r_i = \left(\frac{1}{u_{i,max}} \right)^2$$