Pendulum Control Examples

tags: `control`

- [GitHub Link for code implementation (https://github.com/shengwen-tw/pendulum-control-simulator)](https://github.com/shengwen-tw/pendulum-control-simulator)

Linear Quadratic Regulator (LQR)

State variables:

$x_1 = \theta$

$x_2 = \dot{\theta}$

Control input:

$u = \tau$

$\boxed{\tau : \text{torque}}$

System dynamics:

$\dot{x_1} = x_2$

$\dot{x_2} = -\frac{g}{l} sin(x_1) - \frac{k}{m} x_2 + \frac{1}{ml^2} u$

$\boxed{g : \text{gravitational acceleration}}$

$\boxed{l : \text{rod length}}$

$\boxed{k : \text{air drag coefficient}}$

$\boxed{m : \text{bob mass}}$

Linearized model:

$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$

$$A = \begin{pmatrix} \frac{\partial \dot{x_1}}{\partial x_1} & \frac{\partial \dot{x_1}}{\partial x_2} \\ \frac{\partial \dot{x_2}}{\partial x_1} & \frac{\partial \dot{x_2}}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{g}{l} cos(x_1) & -\frac{k}{m} \end{pmatrix}$$

$$B = \begin{pmatrix} \frac{\partial \dot{x_1}}{\partial u} \\ \frac{\partial \dot{x_2}}{\partial u} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ \frac{1}{ml^2} \end{pmatrix}$$

$$\dot{x} = Ax + Bu$$

Linear Quadratic Regulator:

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

$$\tilde{x} = x - x_d$$

$$u = -K\tilde{x}$$

$$K = R^{-1} B^T X$$

$$\boxed{K : \text{Optimal feedback gain}}$$

CARE (Continuous Algebraic Riccati Equation):

$$A^T X + XA + XBR^{-1} B^T X + C^T Q C$$

$$\boxed{\text{Solve for } X}$$

PID Control (Modeling and S-Domain Analysis)

1. Linearization (Small-Angle Approximation)

For small angles, $\sin(\theta) \approx \theta$. The linearized model becomes:

$$\ddot{\theta} + \frac{k}{m}\dot{\theta} + \frac{g}{l}\theta = \frac{1}{ml^2}u$$

2. Transfer Function (Laplace Domain)

Taking Laplace transform (assuming zero initial conditions):

$$\frac{\Theta(s)}{U(s)} = \frac{1}{ml^2} \cdot \frac{1}{s^2 + \frac{k}{m}s + \frac{g}{l}}$$

With parameters:

- $m = 0.5$, $l = 1$, $g = 9.8$, $k = 1$

The transfer function becomes:

$$G(s) = \frac{2}{s^2 + 2s + 9.8}$$

3. PID Controller Design (S-Domain)

The standard PID controller is:

$$C(s) = K_P + \frac{K_I}{s} + K_D s$$

Example gains:

- $K_P = 30$, $K_I = 10$, $K_D = 5$

Closed-loop system:

$$T(s) = \frac{C(s)G(s)}{1 + C(s)G(s)}$$

4. Performance Metrics (via `stepinfo`)

Metric	Description
Overshoot	Maximum output above the target percentage.
Rise Time	Time taken to rise from 10% to 90% of the target value
Settling Time	Time after which response stays within ±2% of target
Stability	All closed-loop poles in the Left Half Plane (LHP) ⇒ Stable

5. MATLAB Code

```
s = tf('s');
G = 2 / (s^2 + 2*s + 9.8);
C = 30 + 10/s + 5*s;
T = feedback(C*G, 1);
step(T);
stepinfo(T);
pzmap(T);
```