Balancing Robot

Izzy Mones and Heidi Dixon

April 15, 2025

Robot Design

Should have list of all the components. Maybe a picture. Do we need to show circuit stuff?

Model

System Description (Lagrange's Method)

Derive the equations here

$$\begin{aligned}
\dot{x} &= v && (1)\\
\ddot{x} &= \frac{-m^2L^2g\cos(\theta)\sin(\theta) + mL^2(mL\omega^2\sin(\theta) - \delta v) + mL^2u}{mL^2(M + m(1-\cos(\theta)^2))} && (2)\\
\dot{\theta} &= \omega && (3)\\
\dot{\omega} &= \frac{(m+M)mgL\sin(\theta) - mL\cos(\theta)(mL\omega^2\sin(\theta) - \delta v) + mL\cos(\theta)u}{mL^2(M + m(1-\cos(\theta)^2))} && (4)
\end{aligned}$$

where x is the cart position, v is the velocity, θ is the pendulum angle, ω is the angular velocity, m is the pendulum mass, M is the cart mass, L is the pendulum arm length, g is the gravitational acceleration, δ is a friction damping on the cart, and u is the control force applied to the cart.

Linearization

To build a control system for our model we will linearize our system of equations around a fixed point x_r where x_r is the position where the robot is vertical, unmoving and positioned at the origin.

The nonlinear system of differential equations

$$\frac{d}{dt}\boldsymbol{x} = f(\boldsymbol{x}). \tag{5}$$

can be represented as a Taylor series expansion around the point $\boldsymbol{x}_r$.

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_r) + \left.\frac{\boldsymbol{df}}{\boldsymbol{dx}}\right|_{x_r}(\boldsymbol{x} - \boldsymbol{x}_r) + \left.\frac{\boldsymbol{d^2f}}{\boldsymbol{dx^2}}\right|_{x_r}(\boldsymbol{x} - \boldsymbol{x}_r)^2 + \cdots \tag{6}$$

Because $\boldsymbol{x}_r$ is a fixed point, we know that $f(\boldsymbol{x}_r) = 0$. Additionally, this approximation is only accurate in a small neighborhood around $\boldsymbol{x}_r$. In this neighborhood, we can assume that the value of $(\boldsymbol{x} - \boldsymbol{x}_r)$ is small, so higher order terms of this series will go to zero. So a fair estimate of our system is

$$\frac{d}{dt}\boldsymbol{x} \simeq \left.\frac{\boldsymbol{df}}{\boldsymbol{dx}}\right|_{x_r}(\boldsymbol{x} - \boldsymbol{x}_r) \tag{7}$$

where $\frac{\boldsymbol{df}}{\boldsymbol{dx}}\Big|_{x_r}$ is the Jacobian matrix for our system of equations $f(\boldsymbol{x})$ evaluated at the fixed point $\boldsymbol{x}_r$. The Jacobian matrix for our system of equations evaluated at $\boldsymbol{x}_r = [0\ 0\ \pi\ 0]$ is

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{d}{M} & \frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{d}{ML} & -\frac{(m+M)g}{ML} & 0 \end{bmatrix} \tag{8}$$

$$\frac{d}{dt}\boldsymbol{x} \simeq A(\boldsymbol{x} - \boldsymbol{x}_r)$$

LQR

LQG

I'm just testing out how to include code in our document. This is how you do inline code text. `class MyClass` And next is a full code snippet.

```
def loop_iteration():
    global u
    global x

    # estimate state
    dx = (A@(x - xr) + (B*u).transpose() + Kf@(y - C@(x-xr)))[0]
    x = x + dx*dT

    # compute the control value u, and update motor duty cycle
    u = -K@(x - xr)
    motors.run(u * duty_coeff)
```

Experiments

Conclusions