Inverse Pendulum Model and Controller

Aniruddha Gokhale
PSP Active Controls

June 2023

Formulation

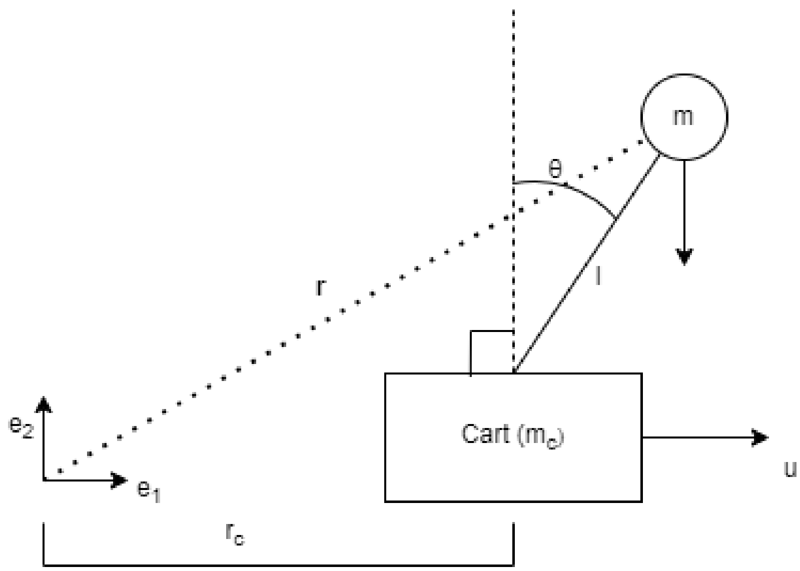

Figure 1: Inverse Pendulum Diagram

Refer to the figure above. A bob of mass m is attached to a stiff rod of length l, which is attached with a pin to a cart of mass m_c, which slides without friction in $\hat{e}_1$. The cart is pulled by an input force u in $\hat{e}_1$ and its position from the origin is r_c. The rod's offset from $\hat{e}_2$ is given by θ. The bob's $\hat{e}$-frame position is given by $\mathbf{r}$.

Kinematics

The position $\mathbf{r}$ is given by

$$\mathbf{r} = (r_c + l\sin\theta)\hat{e}_1 + l\cos\theta\hat{e}_2$$

The earth-frame velocity is therefore

$$\dot{\mathbf{r}} = (\dot{r}_c + \dot{\theta}l\cos\theta)\hat{e}_1 - \dot{\theta}l\sin\theta\hat{e}_2$$

and the acceleration is

$$\ddot{\mathbf{r}} = (\ddot{r}_c + \ddot{\theta}l\cos\theta - \dot{\theta}^2 l\sin\theta)\hat{e}_1 - (\ddot{\theta}l\sin\theta + \dot{\theta}^2 l\cos\theta)\hat{e}_2$$

Dynamics

The bob is subject to the force of gravity and the force of the rod, F_r:

$$m\ddot{\mathbf{r}} = -mg\hat{e}_2 + F_r(\sin\theta\hat{e}_1 + \cos\theta\hat{e}_2)$$

For the cart, normal forces in the $\hat{e}_2$ direction cancel with gravitational forces and vertical forces due to the rod. Therefore, Newton's second law for the cart simplifies to

$$m_c\ddot{r}_c = u - F_r\sin\theta \quad \longrightarrow \quad F_r = \frac{u - m_c\ddot{r}_c}{\sin\theta}$$

Plugging back into the bob's equation gives two equations:

$$\ddot{r}_c + \ddot{\theta}l\cos\theta - \dot{\theta}^2 l\sin\theta = \frac{u}{m} - \frac{m_c\ddot{r}_c}{m}$$

$$-\ddot{\theta}l\sin\theta - \dot{\theta}^2 l\cos\theta = \frac{(u - m_c\ddot{r}_c)\cot\theta}{m} - g$$

Solving these equations yields results for $\ddot{r}_c$ and $\ddot{\theta}$:

$$\ddot{r}_c = \frac{u\sin(\theta) - mg\cos(\theta) + u\cos(\theta)\cot(\theta) + lm\dot{\theta}^2\cos(\theta)^2 + lm\dot{\theta}^2\sin(\theta)^2}{m\sin(\theta) + m_c\sin(\theta) + m_c\cos(\theta)\cot(\theta)}$$

$$\ddot{\theta} = \frac{mg + m_c g - u\cot(\theta) - lm\dot{\theta}^2\cos(\theta) - lm_c\dot{\theta}^2\cos(\theta) + lm_c\dot{\theta}^2\cot(\theta)\sin(\theta)}{l(m\sin(\theta) + m_c\sin(\theta) + m_c\cos(\theta)\cot(\theta))}$$

Control System Design

The state vector is composed as follows:

$$\mathbf{x} = \begin{bmatrix} r_c \\ \dot{r}_c \\ \theta \\ \dot{\theta} \end{bmatrix}$$

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r}_c \\ \frac{u\sin(\theta) - mg\cos(\theta) + u\cos(\theta)\cot(\theta) + lm\dot{\theta}^2\cos(\theta)^2 + lm\dot{\theta}^2\sin(\theta)^2}{m\sin(\theta) + m_c\sin(\theta) + m_c\cos(\theta)\cot(\theta)} \\ \dot{\theta} \\ \frac{mg + m_c g - u\cot(\theta) - lm\dot{\theta}^2\cos(\theta) - lm_c\dot{\theta}^2\cos(\theta) + lm_c\dot{\theta}^2\cot(\theta)\sin(\theta)}{l(m\sin(\theta) + m_c\sin(\theta) + m_c\cos(\theta)\cot(\theta))} \end{bmatrix}$$

The Jacobian of $\dot{\mathbf{x}}$ is taken with respect to $\mathbf{x}$ and u to obtain a linear system. These Jacobians are omitted for brevity.

The Jacobians are then evaluated at critical points of the system, where θ and $\dot{\theta}$ are 0. These become the state-space matrices A and B in the equation

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu$$

The LQR cost matrices Q and R (which is just a scalar) are made according to Bryson's rule. The `lqr` function in `MATLAB` is used to create a gain matrix K, which is then used in the Simulink model.

How to use program

The `MATLAB` live script `InversePendulum.mlx` is the initialization script used to create the equations of motion and parameters used to run the Simulink model.

1. Configure the constants in section 1 of `InversePendulum.mlx` as desired.
2. Configure the critical point, input limits, initial conditions, and cost function parameters in section 3 of `InversePendulum.mlx` as desired.
3. Run `InversePendulum.mlx` to write these parameters to the params file, `params.mat`. This is the file that the Simulink model will reference when running.
4. In the Simulink model, navigate to the MODELING tab and open the Model Explorer. Under the model's name in the tree structure, click on the Model Workspace and click the "Reinitialize from Source" button on the right-hand side. You should see the parameters populate in the main window.
5. In the Simulink model, click Run. You can view the results on the scopes present on the right.