Lab 1

The Inverted Pendulum

Lab Objective: *We will set up the LQR optimal control problem for the inverted pendulum and compute the solution numerically.*

Think back to your childhood days when, for entertainment purposes, you'd balance objects: a book on your head, a spoon on your nose, or even a broom on your hand. Learning how to walk was likely your initial introduction to the inverted pendulum problem.

A pendulum has two rest points: a stable rest point directly underneath the pivot point of the pendulum, and an unstable rest point directly above. The generic pendulum problem is to simply describe the dynamics of the object on the pendulum (called the 'bob'). The inverted pendulum problem seeks to guide the bob toward the unstable fixed point at the top of the pendulum. Since the fixed point is unstable, the bob must be balanced relentlessly to keep it upright.

The inverted pendulum is an important classical problem in dynamics and control theory, and is often used to test different control strategies. One application of the inverted pendulum is the guidance of rockets and missiles. Aerodynamic instability occurs because the center of mass of the rocket is not the same as the center of drag. Small gusts of wind or variations in thrust require constant attention to the orientation of the rocket.

The Simple Pendulum

We begin by studying the simple pendulum setting. Suppose we have a pendulum consisting of a bob with mass m rotating about a pivot point at the end of a (massless) rod of length l. Let $\theta(t)$ represent the angular displacement of the bob from its stable equilibrium. By Hamilton's Principle, the path θ that is taken by the bob minimizes the functional

$$J[\theta] = \int_{t_0}^{t_1} L, \tag{1.1}$$

where the Lagrangian $L = T - U$ is the difference between the kinetic and potential energies of the bob.

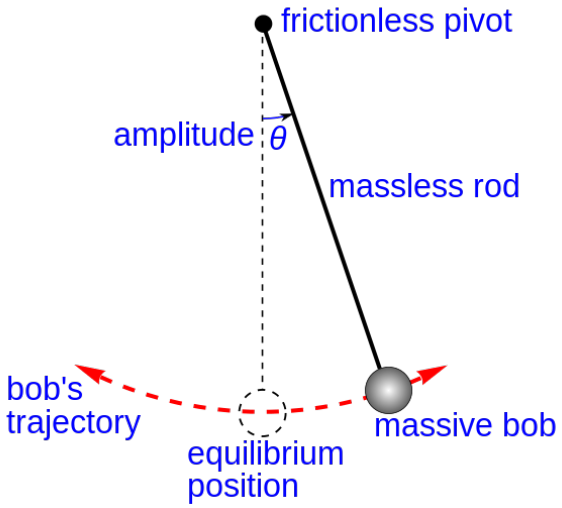


Figure 1.1: The frame of reference for the simple pendulum problem.

The kinetic energy of the bob is given by $mv^2/2$, where v is the velocity of the bob. In terms of θ, the kinetic energy becomes

$$\begin{aligned} T &= \frac{m}{2}v^2 = \frac{m}{2}(\dot{x}^2 + \dot{y}^2), \\ &= \frac{m}{2}((l\cos(\theta)\dot{\theta})^2 + (l\sin(\theta)\dot{\theta})^2), \\ &= \frac{ml^2\dot{\theta}^2}{2}. \end{aligned} \tag{1.2}$$

The potential energy of the bob is $U = mg(l - l\cos\theta)$. From these expressions we can form the Euler-Lagrange equation, which determines the path of the bob:

$$\begin{aligned} 0 &= L_\theta - \frac{d}{dx}L_{\dot{\theta}}, \\ &= -mgl\sin\theta - ml^2\ddot{\theta}, \\ &= \ddot{\theta} + \frac{g}{l}\sin\theta. \end{aligned} \tag{1.3}$$

Since in this setting the energy of the pendulum is conserved, the equilibrium position $\theta = 0$ is only Lyapunov stable. When forces such as friction and air drag are considered $\theta = 0$ becomes an asymptotically stable equilibrium.

The Inverted Pendulum

The Control System

We consider a gift suspended above a rickshaw by a (massless) rod of length l. The rickshaw and its suspended gift will have masses M and m respectively, $M >$

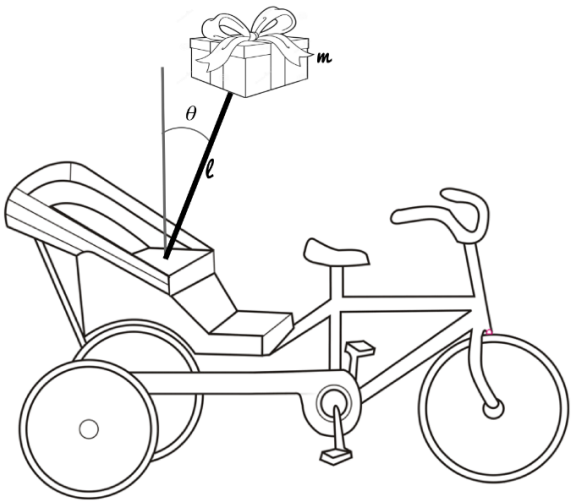


Figure 1.2: The inverted pendulum problem on a mobile rickshaw with a present suspended above.

m. Let θ represent the angle between the gift and its unstable equilibrium, with counterclockwise orientation. Let v_1 and v_2 represent the velocities of the rickshaw and the gift, and F the force exerted on the rickshaw. The rickshaw will be restricted to traveling along a straight line (the x-axis).

By Hamilton's Principle, the path (x, θ) of the rickshaw and the present minimizes the functional

$$J[x, \theta] = \int_{t_0}^{t_1} L, \tag{1.4}$$

where the Lagrangian $L = T - U$ is the difference between the kinetic energy of the present on the pendulum, and its potential energy.

Since the position of the rickshaw and the present are $(x(t), 0)$ and $(x - l\sin\theta, l\cos\theta)$ respectively, the total kinetic energy is

$$\begin{aligned} T &= \frac{1}{2}Mv_1^2 + \frac{1}{2}mv_2^2, \\ &= \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m((\dot{x} - l\dot{\theta}\cos\theta)^2 + (-l\dot{\theta}\sin\theta)^2), \\ &= \dot{x}^2 + l^2\dot{\theta}^2 - 2l\dot{x}\dot{\theta}\cos\theta. \end{aligned} \tag{1.5}$$

The total potential energy is

$$U = mgl\cos\theta - Fx.$$

Since the path (x, θ) of the rickshaw and the present satisfy the Euler-Lagrange

differential equations

$$\begin{aligned} \frac{\partial L}{\partial x} - \frac{d}{dt}\frac{\partial L}{\partial \dot{x}} &= 0, \\ \frac{\partial L}{\partial \theta} - \frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} &= 0, \end{aligned} \tag{1.6}$$

after expanding (1.6) we see that $x(t)$ and $\theta(t)$ satisfy

$$\begin{aligned} F &= (M+m)\ddot{x} - ml\ddot{\theta}\cos\theta + ml\dot{\theta}^2\sin\theta, \\ l\ddot{\theta} &= g\sin\theta + \ddot{x}\cos\theta. \end{aligned} \tag{1.7}$$

At this point we make a further simplifying assumption. If θ starts close to 0, we may assume that the corresponding force F will keep θ small. In this case, we linearize (1.7) about $(\theta, \dot{\theta}) = (0, 0)$, obtaining the equations

$$\begin{aligned} F &= (M+m)\ddot{x} - ml\ddot{\theta}, \\ l\ddot{\theta} &= g\theta + \ddot{x}. \end{aligned}$$

These equations can be further manipulated to obtain

$$\begin{aligned} \ddot{x} &= \frac{1}{M}F - \frac{m}{M}g\theta, \\ \ddot{\theta} &= \frac{1}{Ml}F + \frac{g}{Ml}(M+m)\theta. \end{aligned} \tag{1.8}$$

We will now write (1.8) as a first order system. Making the assignments $x_1 = x$, $x_2 = x_1'$, $\theta_1 = \theta$, $\theta_2 = \theta_1'$, letting $u = F$ represent the control variable, we obtain

$$\begin{bmatrix} x_1 \\ x_2 \\ \theta_1 \\ \theta_2 \end{bmatrix}' = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{mG}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g}{Ml}(M+m) & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \theta_1 \\ \theta_2 \end{bmatrix} + u \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml} \end{bmatrix},$$

which can be written more concisely as

$$z' = Az + Bu.$$

Problem 1. Write a function that returns the matrices A, B, Q, and R given above. Let $g = 9.8$ m/s^2.

```
def linearized_init(M, m, l, q1, q2, q3, q4, r):
    '''
    Parameters:
    ----------
    M, m: floats
          masses of the rickshaw and the present
    l   : float
          length of the rod
    q1, q2, q3, q4, r : floats
         relative weights of the position and velocity of the rickshaw, ↩
             the
         angular displacement theta and the change in theta, and the ↩
```

```
            control

    Return
    -------
    A : ndarray of shape (4,4)
    B : ndarray of shape (4,1)
    Q : ndarray of shape (4,4)
    R : ndarray of shape (1,1)
    '''
    pass
```

The infinite time horizon LQR problem

We consider the cost function

$$\begin{aligned} J[z] &= \int_0^\infty (q_1 x_1^2 + q_2 x_2^2 + q_3 \theta_1^2 + q_4 \theta_2^2 + r u^2)\, dt \\ &= \int_0^\infty z^T Q z + u^T R u \, dt \end{aligned} \tag{1.9}$$

where q_1, q_2, q_3, q_4, and r are nonnegative weights, and

$$Q = \begin{bmatrix} q_1 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 \\ 0 & 0 & q_3 & 0 \\ 0 & 0 & 0 & q_4 \end{bmatrix}, R = \begin{bmatrix} r \end{bmatrix}.$$

The optimal control problem (1.9) is an example of a Linear Quadratic Regulator (LQR), and is known to have an optimal control $\tilde{u}$ described by a linear state feedback law:

$$\tilde{u} = -R^{-1} B^T P \tilde{z}.$$

Here P is a matrix function that satisfies the Riccati differential equation (RDE)

$$\dot{P}(t) = PA + A^T P + Q - PBR^{-1}B^T P.$$

Since this problem has an infinite time horizon, we have $\dot{P} = 0$. Thus P is a constant matrix, and can be found by solving the algebraic Riccati equation (ARE)

$$PA + A^T P + Q - PBR^{-1}B^T P = 0. \tag{1.10}$$

The evolution of the optimal state vector $\tilde{z}$ can then be described by [1]

$$\dot{\tilde{z}} = (A - BR^{-1}B^T P)\tilde{z}. \tag{1.11}$$

[1] See Calculus of Variations and Optimal Control Theory, Daniel Liberzon, Ch.6

Problem 2. Write the following function to find the matrix P. Use `scipy.optimize.root`. Since `root` takes in a vector and not a matrix, you will have to reshape the matrix P before passing it in and after getting your result, using `np.reshape(16)` and `np.reshape((4,4))`.

```
def find_P(A,B,Q,R):
    '''
    Parameters:
    ----------
    A, B, Q : ndarrays of shape (4,4)
    R       : ndarray of shape (1,1)

    Returns
    -------
    P       : the matrix solution of the Riccati equation
    '''
    pass
```

Using the values

```
M, m = 23., 5.
l = 4.
q1, q2, q3, q4 = 1., 1., 1., 1.
r = 5.
```

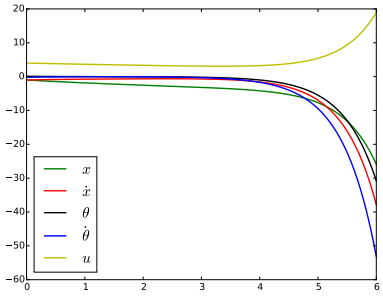
compute the eigenvalues of $A - BR^{-1}B^TP$. Are any of the eigenvalues positive? Consider differential equation (1.11) governing the optimal state $\tilde{z}$. Using this value of P, will we necessarily have $\dot{\tilde{z}} \to 0$?

Notice that we have no information on how many solutions (1.10) possesses. In general there may be many solutions. We hope to find a unique solution P that is *stabilizing*: the eigenvalues of $A - BR^{-1}B^TP$ have negative real part. To find this P, use the function `solve_continuous_are` from `scipy.linalg`. This function is designed to solve the continuous algebraic Riccati equation.

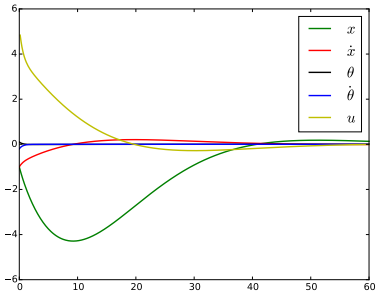
Problem 3. Write the following function that implements the LQR solution described earlier. For the IVP solver, you can use your own or you may use the function `ode` from `scipy.integrate`.

```
def rickshaw(tv,X0,A, B, Q, R_inv, P):
    '''
    Parameters:
    ----------
    tv : ndarray of time values, with shape (n+1,)
    X0 :
    A, B, Q : ndarrays of shape (4,4)
    R_inv   : ndarray of shape (1,1), inverse of R
    P       : ndarray of shape (4,4)

    Returns
```



P is found using `scipy.optimize.root`.



P is found using `solve_continuous_are`.

Figure 1.3: The solutions of Problem 4.

```
    -------
    Z : ndarray of shape (n+1,4), the state vector at each time
    U : ndarray of shape (n,), the control values
    '''
```

Problem 4. Test the function made in Problem (3) with the following inputs:

```
M, m = 23., 5.
l = 4.
q1, q2, q3, q4 = 1., 1., 1., 1.
r = 10.
tf = 15
X0 = np.array([-1, -1, .1, -.2])
```

Use both `scipy.optimize.root` and `solve_continuous_are` to find the matrix P. Compare your results. The results are plotted in Figure 1.3.

Problem 5. Consider the following inputs:

```
M, m = 23., 5.
l = 4.
q1, q2, q3, q4 = 1., 1., 1., 1.
r = 10.
tf = 60
X0 = np.array([-1, -1, .1, -.2])
```

Vary the entries of `X0` responsible for $\theta(0)$ and $\dot{\theta}(0)$ to determine the sensitivity

of the control u to the initial conditions. What initial conditions lead to a reasonable, physical control u?