Project Outline

Zac Yauney, Andrew Williams

March 14, 2022

As a reminder, the purpose of this project is to solve optimal control to allow a rocket to land upright with minimal velocity and expending minimal fuel. Control is given by choosing how much to fire both the vertical and rotational thrusters at each point in time.

We will be modeling this with discrete time.

1 State Equations

We set this problem up for LQR Optimization with the following scheme

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ x' \\ y' \\ \theta \\ T \end{bmatrix} \quad \mathbf{u}' = \begin{bmatrix} p \\ r \\ g \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 0 & dt & 0 & 0 & 0 \\ 0 & 0 & 0 & dt & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cos(\theta) \\ 0 & 0 & 0 & 0 & 0 & \sin(\theta) \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & dt \\ 0 & dt & 0 \\ dt & 0 & 0 \end{bmatrix}$$

$$\mathbf{x}' = A\mathbf{x} + B\mathbf{u}$$

and we have the following definitions for our variables:

$$x = \text{x position}$$

$$y = \text{y position}$$

$$\theta = \text{Ship's angle relative to vertical}$$

$$T = \text{Current Main Engine Power}$$

$$p = \text{Main Engine Thruster Control}$$

$$r = \text{Rotational Thruster Control}$$

$$g = \text{Gravitational Acceleration Constant}$$

We also have boundary conditions

$$x(t_f) = 0$$

$$y(t_f) = 0$$

The initial conditions vary from iteration to iteration. We start in a random location with varying amounts of fuel.

2   Cost Function

$$J[u] = \int_0^{t_f} \mathbf{u}^\intercal R(t)\mathbf{u}\,dt + \mathbf{x_f}^\intercal M \mathbf{x_f}$$

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

which means $Q(t) = 0$

3   Approach

We will want to use LQR optimization using the given matrices. The biggest challenge will be linearizing the angle dependent terms.

Once we have the linearization figured out, we will take the initial conditions and numerically solve for the optimal control, which we will feed into a simulator program that will let us watch the progress of our algorithm in real time.