Problem Statement and Solution Ideas

Group Members: Dallan Olsen, Laren Edwards, Riley Gabrielson, Calix Barrus

1. Premise

Our foundation is a robot with 4 independent wheels, and our goal is to solve optimal avoidance paths using the 4 wheels as our control. We feel that this extension to the standard object avoidance problem provides more direct application to real world robotics problems.

2. Cost Function

We have for our cost functional the equation J[u] =0tfc1 ds+ 0tfc2(x"2 + y"2) + c3 C(x, y)dt +c4tf Where C(x,y) is given as in the Obstacle avoidance lab. This cost functional allows us to penalize time, path length, acceleration, as well as the ability to impose a stiff penalty for colliding with (or getting too close to) the obstacle.

3. State Equations

The evolution of our state is given by

$$\boldsymbol{x}' = H \begin{bmatrix} x \\ y \\ \theta \\ x' \\ y' \\ \theta' \end{bmatrix} + F \begin{bmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{\phi}(\boldsymbol{u}) \end{bmatrix}$$

$$\boldsymbol{x}(t_f) = [x_f, y_f, \ldots]^T$$

Where

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, F = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

This basically states that first derivatives are related to themselves and the second derivatives are controlled by the control variable.

The phi function is a result of solving the equation of motion, and describes how the 4 controls (motors 1,2,3,4) influence x', y', and theta'. In order to solve this problem with LQR, we may need to linearize phi.

4. Plan to Solve

i. Derive and solve equations of motion to get ϕ.
ii. Solve for optimal path using LQR or a numerical scheme. This will involve making coming up with the LQR equations and solving for the optimal path.
iii. Come up with jupyter code and plots of states and controls.

iv. Plug situation into physics engine.
   a. Use PID to stick to optimal path in engine.
v. Attempt recreate simulation with physical robots.
   a. Work out how the sensors work and getting them to feed info to our code.