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 $\phi$.
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.

Compute path w/ finite horizon LQR, recompute path every little bit.

## 5. Wall Following Scenario

In this scenario, we will try to control a robot car and have it follow a wall while staying a fixed distance away from the wall.

5.1. **LQR Formulation.** For our state, we will consider the distance between the car and the center line to be following that is perpendicular to the wall ($y$). We will also track the bearing of the car ($\theta$), the velocity of the right wheels ($V_r$), and the velocity of the left wheels ($V_\ell$).

Our state equation is given as

$$\boldsymbol{x} = \begin{bmatrix} y \\ \theta \\ V_r \\ V_\ell \end{bmatrix}, \boldsymbol{u} = \begin{bmatrix} V_r^c \\ V_\ell^c \end{bmatrix}$$

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

$$A = \begin{bmatrix} 0 & V_{nom} & 0 & 0 \\ 0 & 0 & \frac{1}{2}d & -\frac{1}{2}d \\ 0 & 0 & -a & 0 \\ 0 & 0 & 0 & -a \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ a & 0 \\ 0 & a \end{bmatrix}$$

$$J[u] = \int_{t_i}^{t_i+\Delta t} \boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \, dt$$

$$Q = \begin{bmatrix} q_1 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, R = rI$$

where above, $V_r^c, V_\ell^c$ are the left and right commanded wheel velocities, $d$ is the diameter of the robot, distance between the centers of the wheels, $a$ is a delay, $q_1$ and $q_2$ are the penalties for being away from the center line, and having a not straight bearing, and $r$ is a penalty for high wheel commanded velocity.

Note here that $y' = v\sin(\theta)$, but since $\theta \approx 0$ so $\sin(\theta) \approx \theta)$ and $v \approx v_{nom}\theta$.

Note that for boundary conditions, we have that $\boldsymbol{x}(0) = \boldsymbol{x}_0, V_r(t_i + \Delta t) = V_\ell(t_i + \Delta t) = V_{nom}$