F1TENTH Autonomous Racing **(Due Date:)**

Lab 3: Wall Following

Instructor: INSTRUCTOR *Name:* STUDENT NAME, *StudentID:* ID

Course Policy: Read all the instructions below carefully before you start working on the assignment, and before you make a submission. All sources of material must be cited. The University Academic Code of Conduct will be strictly enforced.

1 Learning outcomes

The goal of this lab is to ...

- PID Controllers
- Driving the car autonomously via Wall Following

2 Review of PID in the time domain

In the lecture we saw PID in the frequency domain, since that brings out most clearly the effects of the various gains and why we might want to add, say, a derivative term. Here we look at PID in the time domain, in relation to the task of wall-following.

A PID controller is a way to maintain certain parameters of a system around a specified set point. PID controllers are used in a variety of applications requiring closed-loop control, such as in the VESC speed controller on your car.

The general equation for a PID controller in the time domain, as discussed in lecture, is as follows:

$$u(t) = K_p\, e(t) + K_i \int_0^t e(t')\, dt' + K_d \frac{d}{dt}(e(t))$$

Here, K_p, K_i, and K_d are constants that determine how much weight each of the three components (proportional, integral, derivative) contribute to the control output $u(t)$. $u(t)$ in our case is the steering angle we want the car to drive at. The error term $e(t)$ is the difference between the set point and the parameter we want to maintain around that set point.

3 Wall Following

In the context of our car, the desired distance to the wall should be our set point for our controller, which means our error is the difference between the desired and actual distance to the wall. This raises an important question: how do we measure the distance to the wall, and at what point in time? One option would simply be to consider the distance to the right wall at the current time t (let's call it D_t). Let's consider a generic orientation of the car with respect to the right wall and suppose the angle between the car's x-axis and the wall is denoted by α. We will obtain two laser scans (distances) to the wall: one at an angle θ ($0 < \theta \leq 70$ degrees), and another at an angle of 0 degrees relative to the car's x-axis. Suppose these two laser scans return distances a and b, respectively.

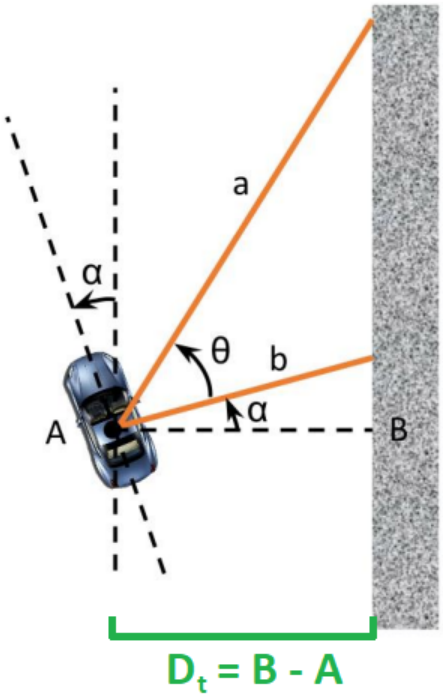

Figure 1: Distance and orientation of the car relative to the wall

Using the two distances a and b from the laser scan, the angle θ between the laser scans, and some trigonometry, we can express α as

$$\alpha = \tan^{-1}\left(\frac{a\cos(\theta) - b}{a\sin(\theta)}\right) \tag{3.1}$$

We can then express D_t as

$$D_t = b\cos(\alpha)$$

to get the current distance between the car and the right wall. What's our error term $e(t)$, then? It's simply the difference between the desired distance and actual distance! For example, if our desired distance is 1 meter from the wall, then $e(t)$ becomes $1 - D_t$.

However, we have a problem on our hands. Remember that this is a race: your car will be traveling at a high speed and therefore will have a non-instantaneous response to whatever speed and servo control you give to it. If we simply use the current distance to the wall, we might end up turning too late, and the car may crash. Therefore, we must look to the future and project the car ahead by a certain lookahead distance (let's call it L). Our new distance D_{t+1} will then be

$$D_{t+1} = D_t + L \sin(\alpha)$$

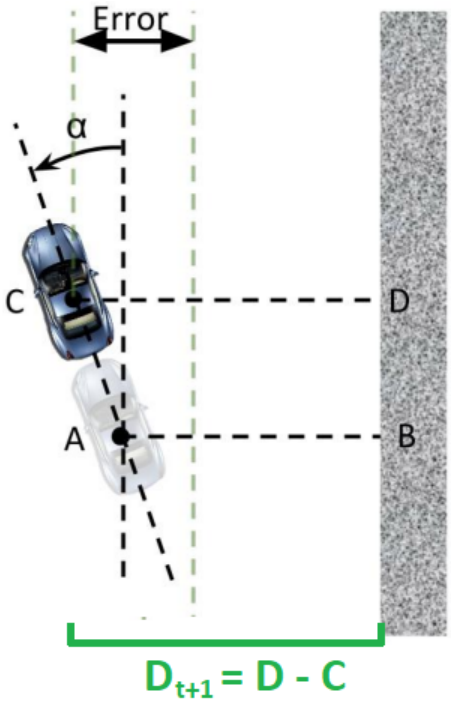

Figure 2: Finding the future distance from the car to the wall

We're almost there. Our control algorithm gives us a steering angle for the VESC, but we would also like to slow the car down around corners for safety. We can compute the speed in a step-like fashion based on the steering angle so that as the angle exceeds progressively larger amounts, the speed is cut in discrete increments. For this lab, we would like you to implement the following speed control algorithm:

- If the steering angle is between 0 degrees and 10 degrees, the car should drive at 1.5 meters per second.
- If the steering angle is between 10 degrees and 20 degrees, the speed should be 1.0 meters per second.
- Otherwise, the speed should be 0.5 meters per second.

So, in summary, here's what we need to do:

1. Obtain two laser scans (distances) a and b, with b taken at 0 degrees and a taken at an angle θ $(0 < \theta \leq 70)$
2. Use the distances a and b to calculate the angle α between the car's x-axis and the right wall.

3. Use α to find the current distance D_t to the car, and then α and D_t to find the estimated future distance D_{t+1} to the wall.
4. Run D_{t+1} through the PID algorithm described above to get a steering angle.
5. Use the steering angle you computed in the previous step to compute a safe driving speed.
6. Publish the steering angle and driving speed to the VESC.

4 Implementation

Implement left-wall following to make the car drive autonomously around the Levine Hall map. You can implement this node in either C++ or Python but the skeleton code is only in Python. You can download it from https://github.com/f1tenth/f1tenth_labs/tree/master/lab3. You will only have to edit `wall_follow.py`

5 Deliverables and Submission

Submit the following as `studentname_lab3.zip` (replace studentname with your name):

1. Your package named `studentname_wallfollow.zip` including the wall following node. **Make sure it compiles before you submit after changing the package name.**
2. Make a youtube video of wall following around the Levine Loop in the simulator add this link to a text file named `studentname_lab3_video.txt`

6 Grading

6.1 Rubric

Topics	**Points**
Compilation	10
Implemented PID	40
Tuned PID	40
Video	10
Total	100