Final Race and Fast Obstacle Avoidance in TESSE

MIT Robotic Science + System (6.141) Team 11

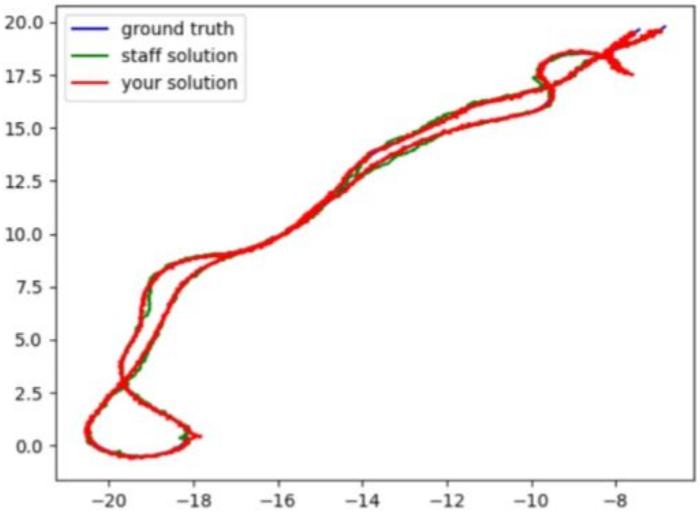
Self driving cars may increase future road safety

- 38,000 people are estimated to be killed in car accidents every year and an additional 4.4 million are seriously injured
- Self driving vehicles can often sense obstacles more effectively than humans
- A collaborative self driving network can be created to avoid accidents and reduce fatalities

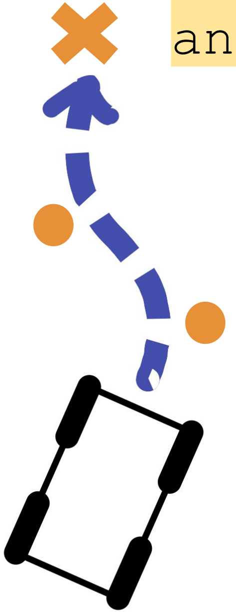
Collaborative, self-driving network

Now that we've solved major autonomous vehicle issues, we can now test the capabilities of our final system!

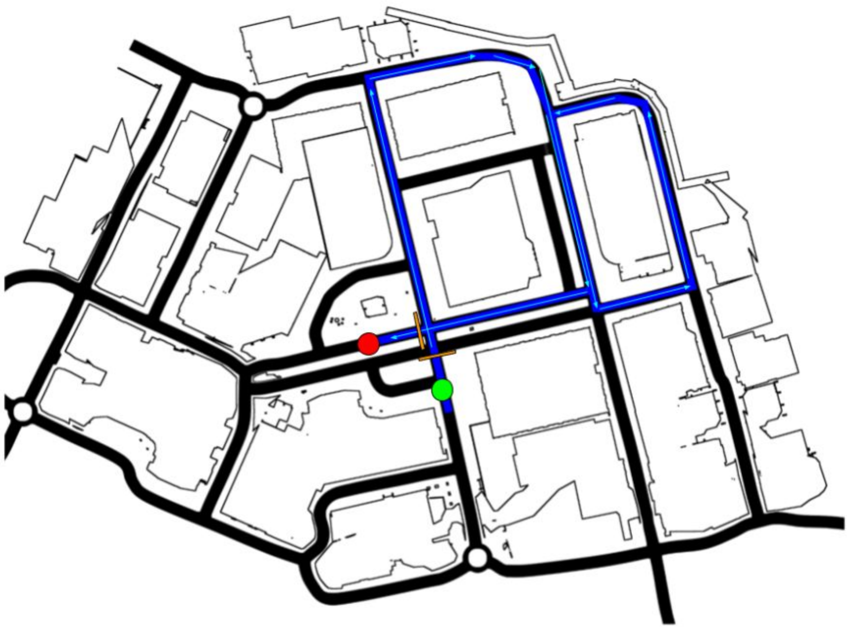
Lab 5 - Localization



Lab 6 - Path Planning and Following



Challenge 1: Final Race



Challenge Goals

- Complete a lap around Windridge City
- Minimize time taken to complete lap
- Complete lap with no collisions

Methodology

- Design lap trajectory
- Use ground truth localization
- Use pure pursuit path following
- Track collisions

Challenge 2: Fast Obstacle Avoidance

Challenge Goals

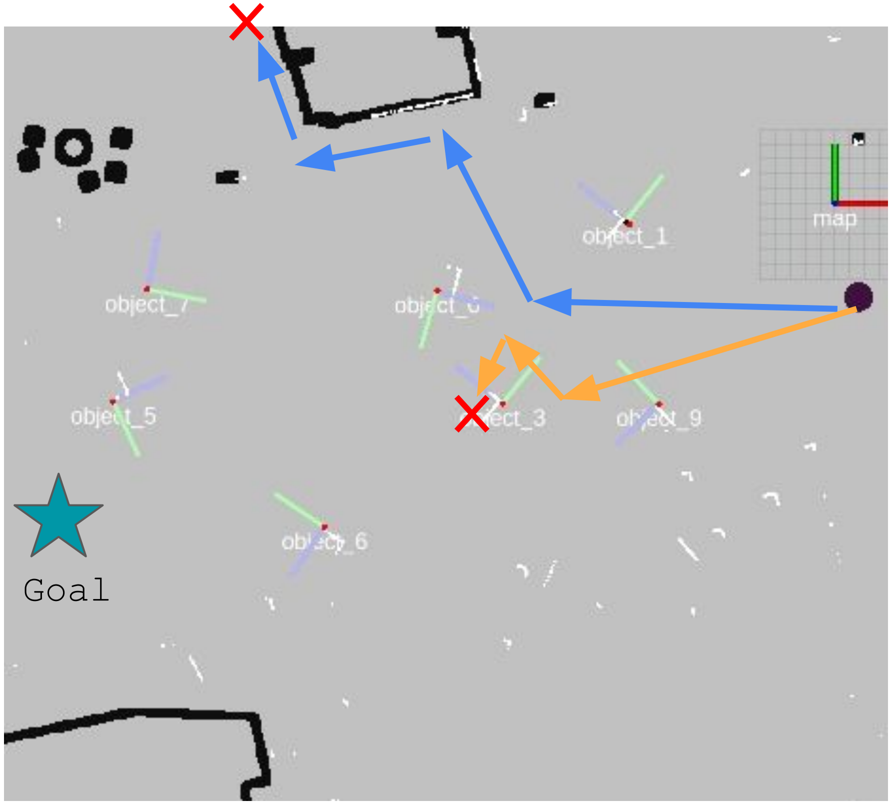
- Traverse obstructed road
- Minimize time taken to navigate road
- Minimize collisions
- Implementation robust to varying number of obstacles

Methodology

- Use TESSE LiDAR
- Path plan around obstacles
- Track collisions

First, obstacle avoidance.

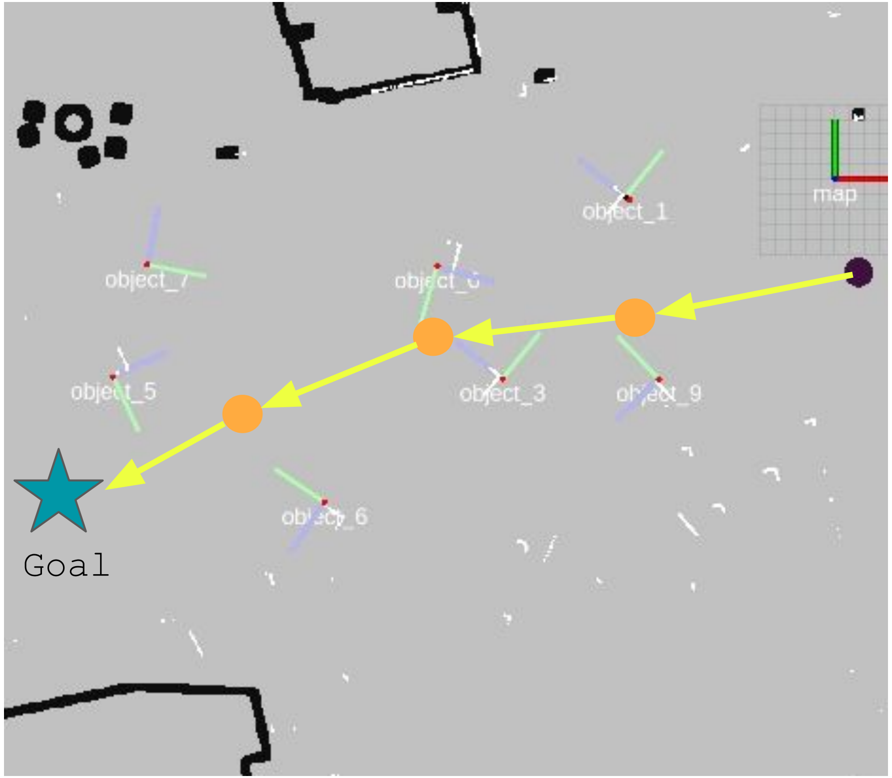
Challenge 2: Obstacle Avoidance



Initially, we tried a few simple algorithms.

- **Pure Obstacle Avoidance:**
  - If obstacle in front, swerve, else drive straight
  - Problem: Drives off track
- **Goal Sensitive Obstacle Avoidance:**
  - If obstacle in front, swerve, else drive towards goal point
  - Problem: Crashes into obstacles it previously swerved to avoid

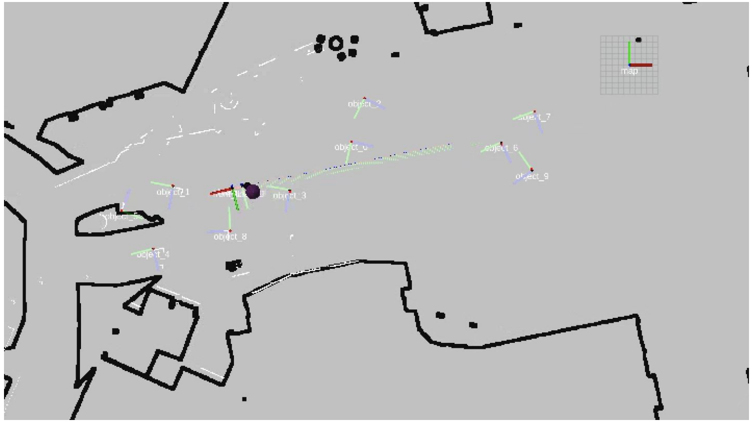
Real-time path planning was the most successful approach

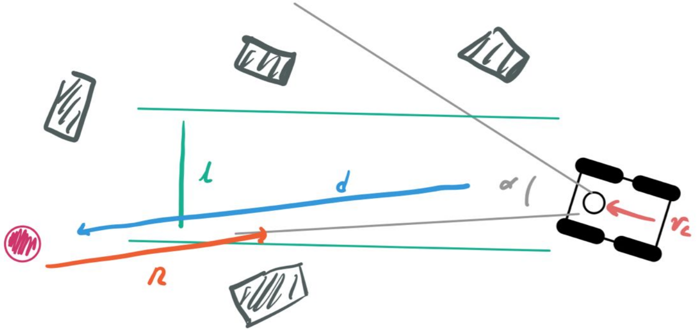


We ended up using a real-time path planning algorithm:

1. Given a lidar reading, **calculate all valid waypoints** at a certain lookahead distance by looking at **"open lanes."**
2. Chose waypoint **closest** to goal point.
3. Drive some distance towards waypoint, then **repeat**.

Parameters were tuned to optimize our solution





Parameters to tune and how we tuned them

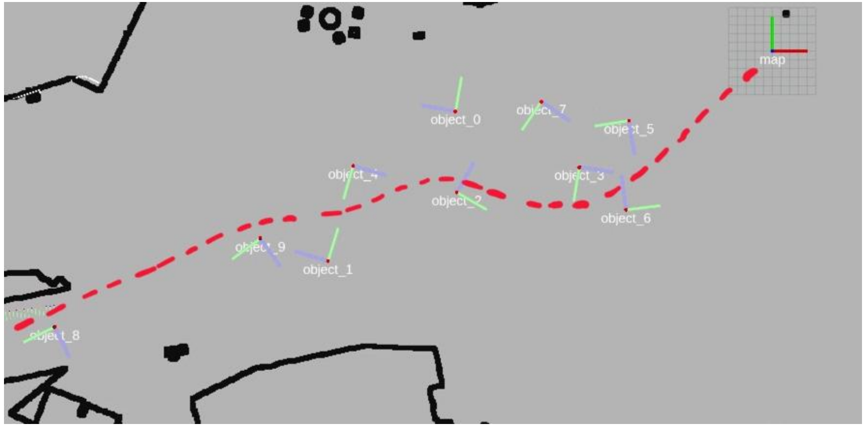
- **Vc → commanded speed** of the vehicle in m/s
- **alpha → lidar angle** to search through for a clear "lane"
- **l → lane width** to search for
- **d → look-ahead distance** (how far should the lane be clear)
- **R → replan distance** (how far from the waypoint should we re-plan the waypoint)
- **Cd → angle damping coefficient** (damps output of pure-pursuit steering angle As = As_pp/Cd)

Other things we changed: location of the **goal point**

Tuning Parameters

- High-speed (for fast time)
- Large look-ahead (to account for the high speed)
- Quick re-plan (to re-check as fast as possible)
- Medium lane-width (so car can sneak through tight spaces but not hit everything)
- Small alpha-value (constrain the car from turning too sharply)
- Medium damping-coefficient (prevents problematic oscillations while allowing the car to react quickly)

Challenge 2: Obstacle Avoidance Results



Time Achieved: 5.47 s
~92.5m track
V_{avg} ~ 17m/s

**No graphs are presented as we felt graphs are not very indicative of performance of the controller

Evaluation Metrics

- Based on **speed** as well as **consistency** of algorithm
- Visual inspection of performance as well

Speed

- Speed on VDI was evaluated **based on the input parameter Vc**
- Speed in competition was evaluated based on the fastest times we achieved

Consistency

- With a set of parameters **ran the race a set number of times and counted successes + failures**

10 Obstacles: 25 Runs → 18 Successful (72% success rate)

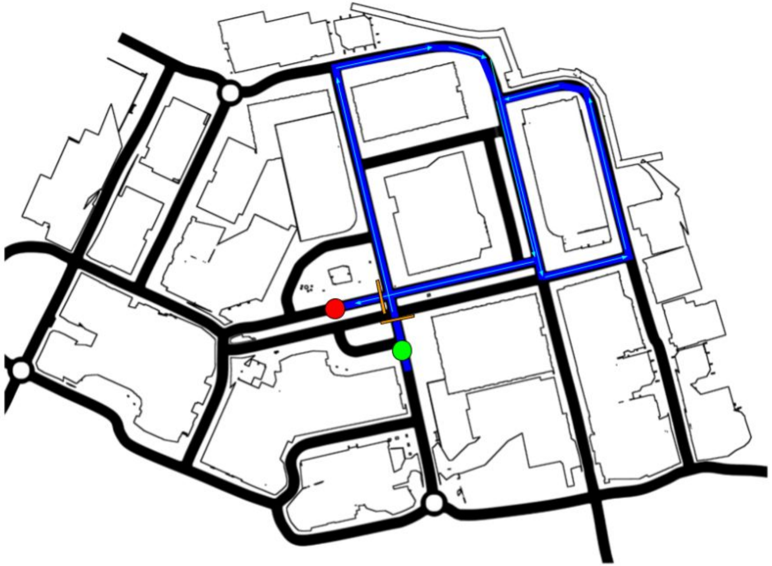
Visual Inspection

- Notice vehicle **can navigate complex paths (see pic)**
- Used collisions to re-tune + note failure cases to make controller better

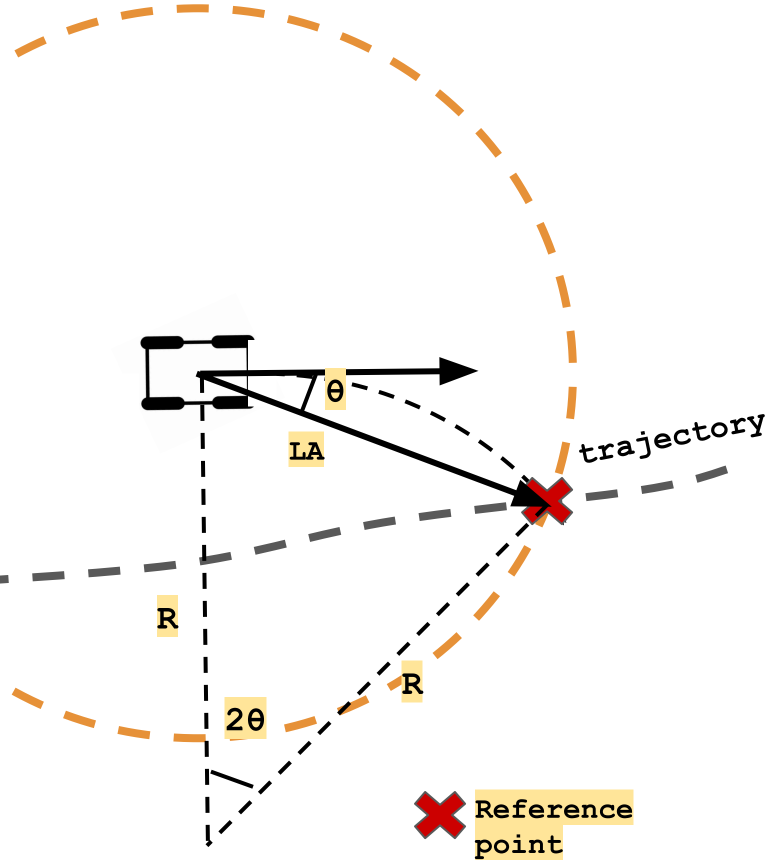
Fast Obstacle Avoidance Conclusions

- For **high speed obstacle avoidance**, complicated **dynamic path planning *may* be needed** to achieve good performance.
- It is **important to choose *available* parameters correctly** to be able to accurately tune a controller for a desired performance.
- Our obstacle-avoidance algorithm is **very expensive to run**, there may be ways to make lighter versions of the algorithm in the future.
- We successfully met our goals and **navigated the obstructed road in 5.47s with no collisions**

Next, the race.



The Basics Behind Pure Pursuit



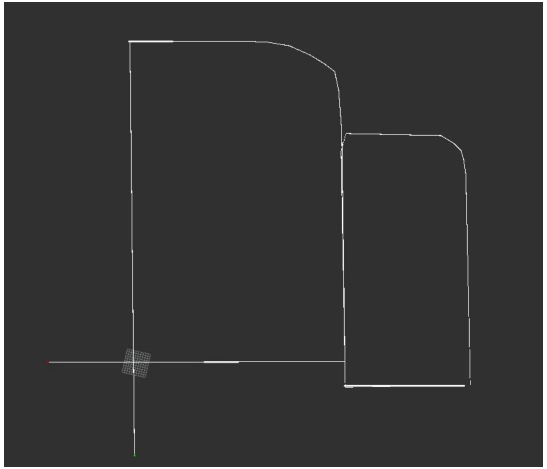
Used to closely and smoothly track a reference trajectory

To use:

- Select a reference point on the desired trajectory a fixed distance LA from the car (LA = look ahead distance)
- Choose a steering angle to hit the reference point if the steering was kept constant

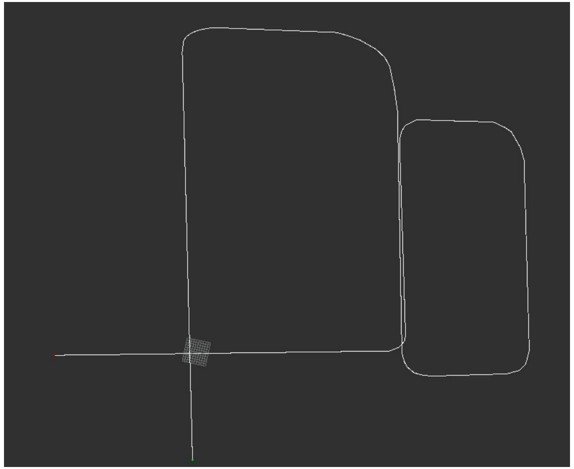
Our path evolved throughout testing

Path 1



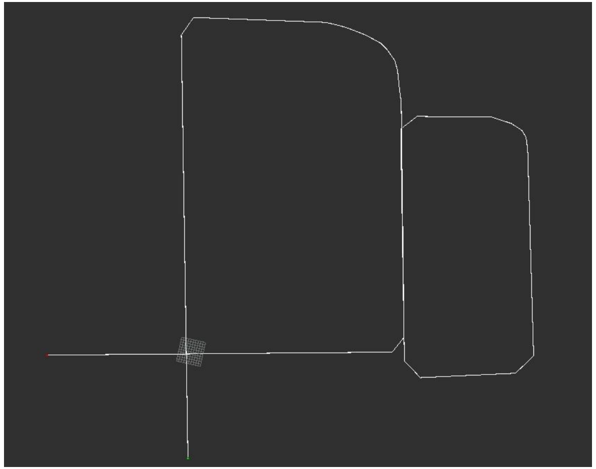
- Straight and long segments allow for higher maximum velocity
- 90 degree angles lead to inaccurate turns and collisions

Path 2



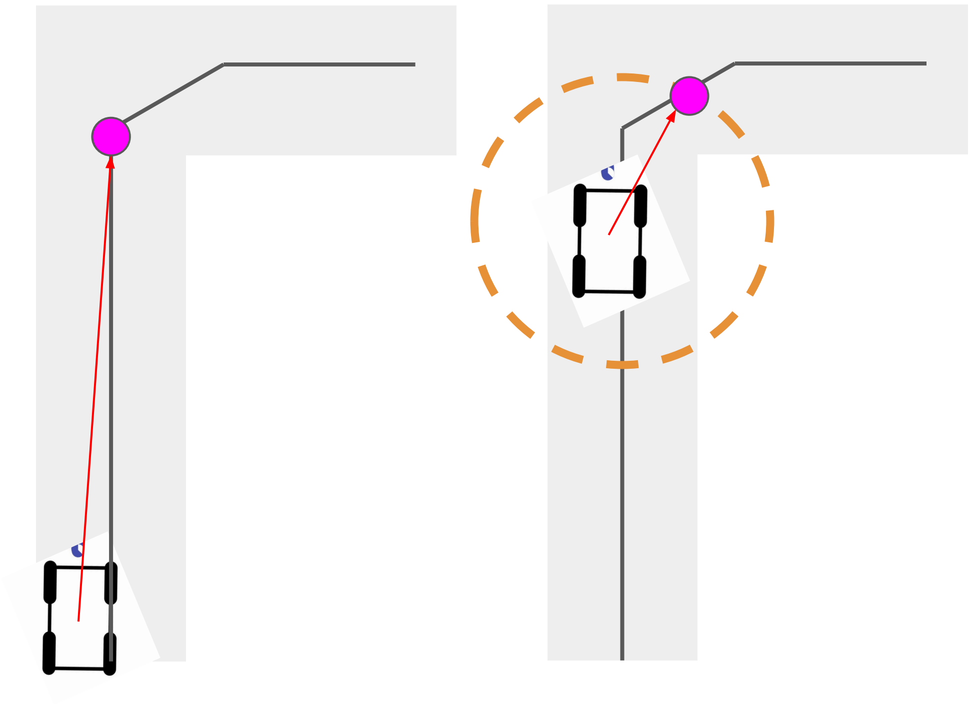
- Implementing racing lines
- Making the path align more closely with our expected trajectory of the car

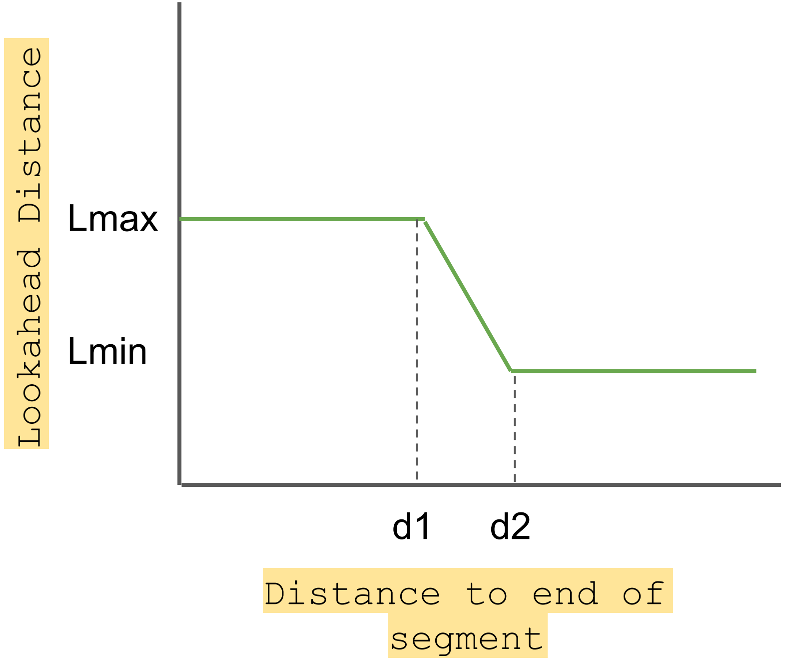
Path 3



- Altered turns that were taken too wide to have 2 points and more closely cut corners

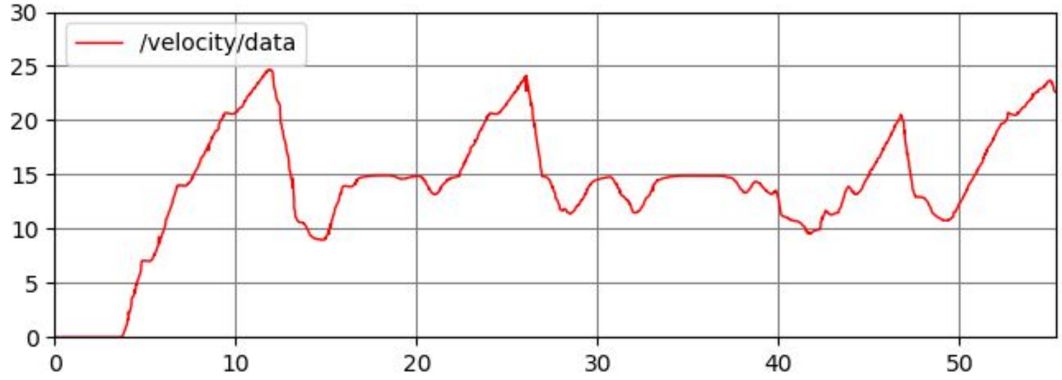
Collisions were minimized with Variable Lookahead Distances

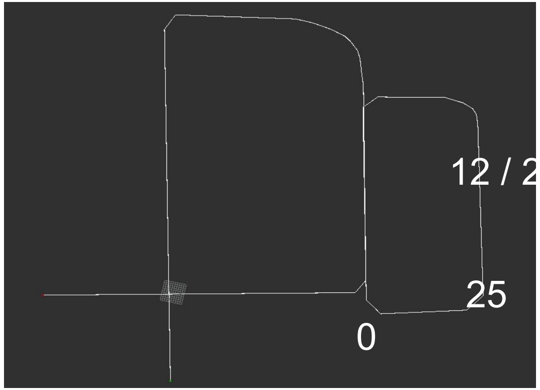


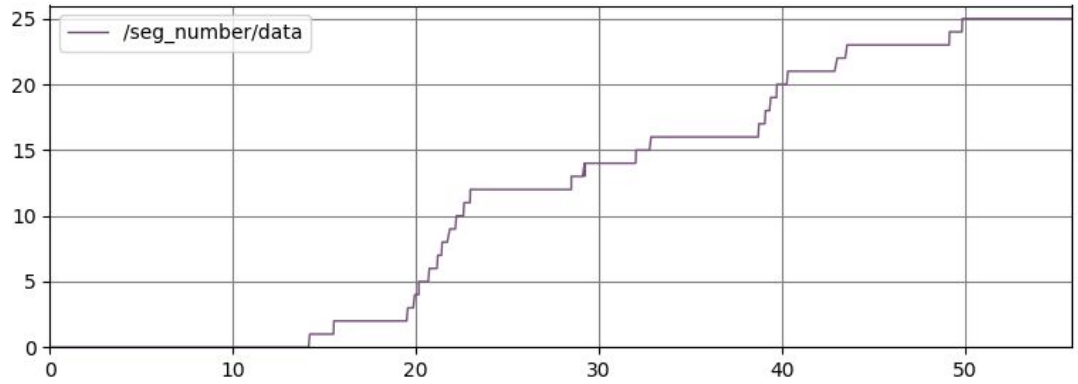


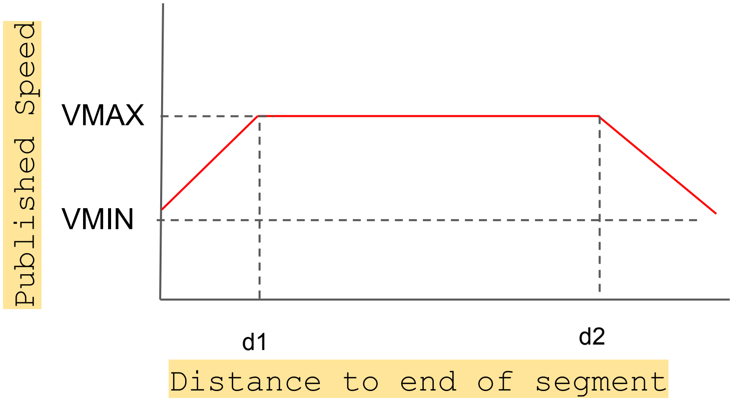
**Lmax, Lmin, d1, and d2 were tuned to the needs of specific corners in the race

Average Speed was maximized with Variable Velocities

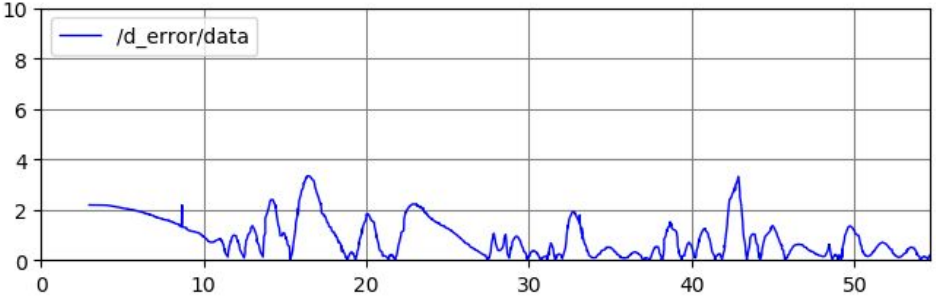


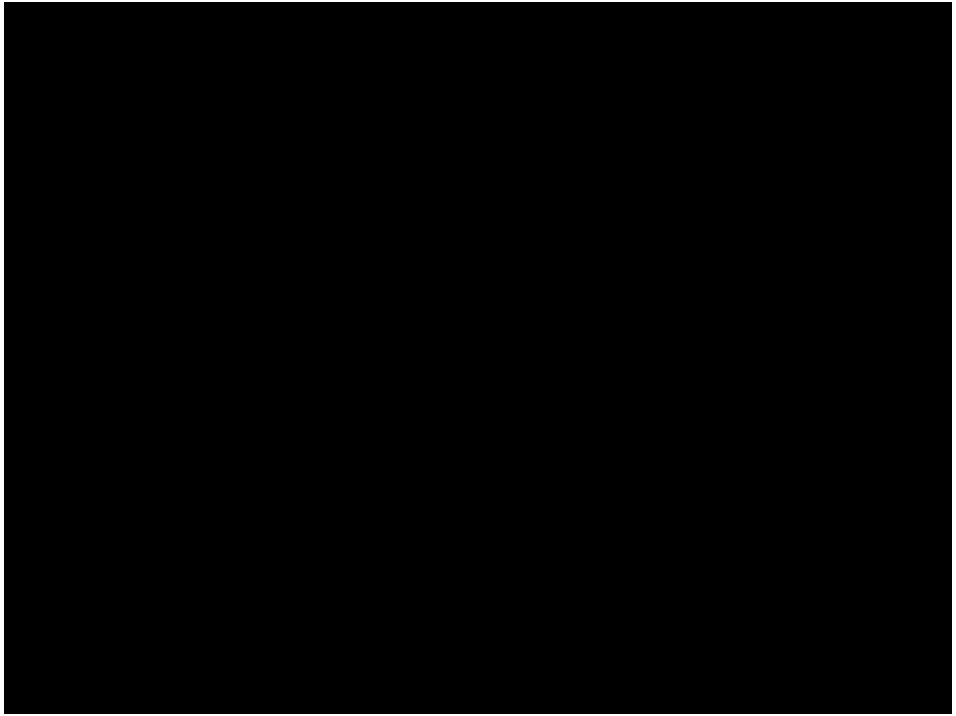






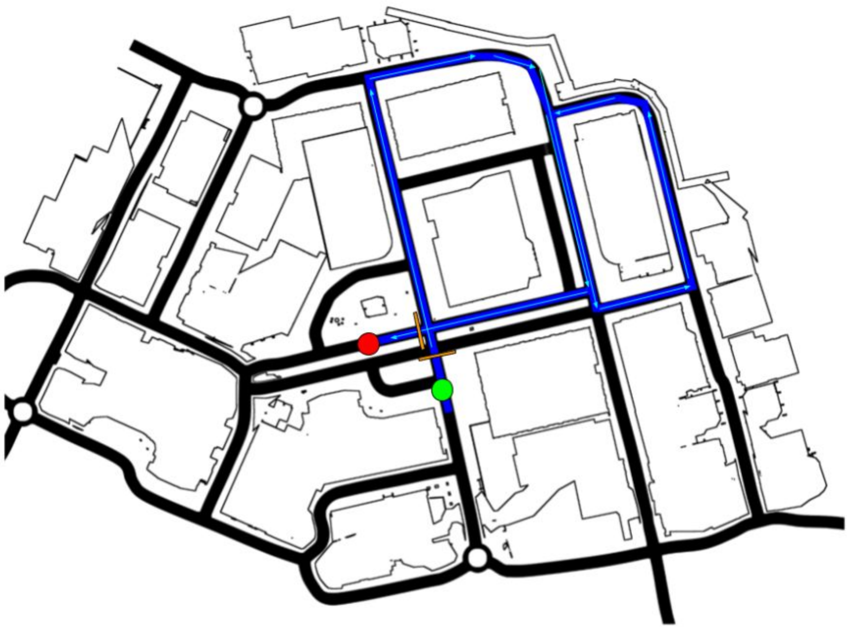
Final Race in Action





- VDI lap time: **47.6s**;
  # of Collisions: **0**
- Final Race time: **49.32s**;
  # of Collisions: **0**

Final Race Conclusions - We met our goals!



Challenge Goals Met

- Completed a lap around Windridge City
- Completed lap in 49 seconds
- Completed lap with no collisions

Thank you!

Questions?

Ka-CHOW!

~ lightning mcqueen

Also, thank you to all Instructors + Course Staff for a great semester!!! :)