RSS Final Challenge

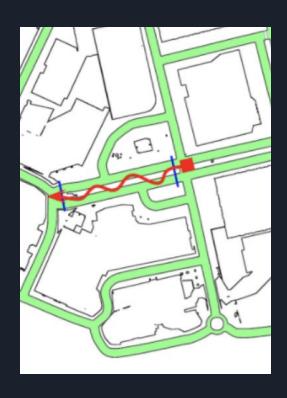
Team Bird-Planes

Joshua, Isaac, Lilly, Mario

Final Challenge



Final Race



Obstacle Avoidance

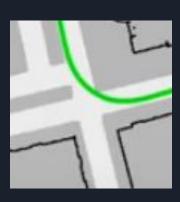
Challenge 1: Final Race

Building the Trajectory

Objectives:

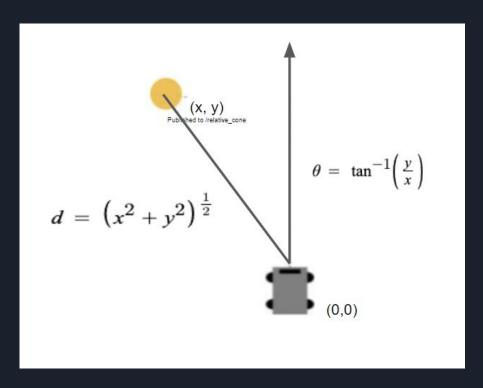
- 1. Safety first!
- 2. Tune pure-pursuit target speed / path





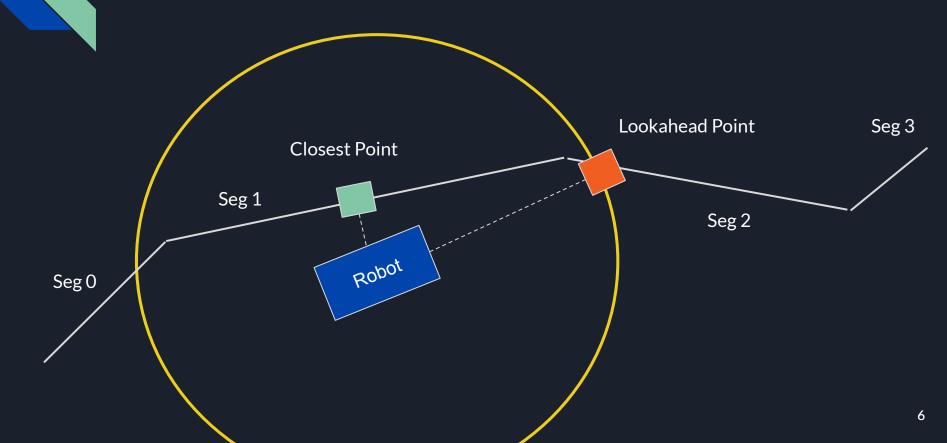
- ->Avoided clipping corners
- -> Entered turns slightly wider and "apexed" around the corner

Pure Pursuit follows a Transient Goal



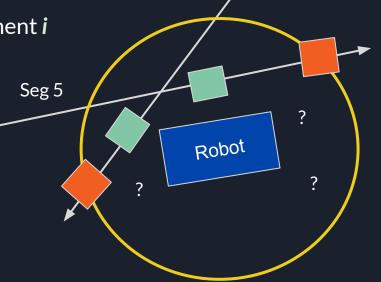
$$\delta = \tan^{-1} \left(\frac{2L \sin \eta}{L_1} \right)$$

Transient Goal is chosen along Trajectory



Modifications

- 1. Loops and Intersections
 - Where can closest point be?
 - Must be on most recent segment i
 - Can be on segment *i+1*
 - Nowhere else!

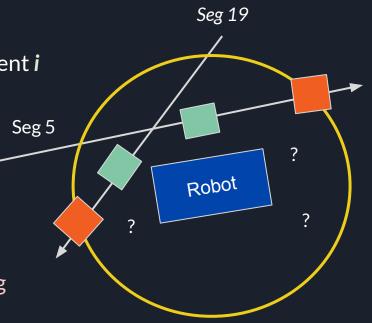


Seg 19

Modifications

- 1. Loops and Intersections
 - Where can closest point be?
 - Must be on most recent segment i
 - Can be on segment i+1
 - Nowhere else!

- 2. Cross-Platform Stability
 - o 2D Sim: Nominal
 - 3D Sim: Oscillating and drifting
 - Solution: P Controller
 - i. ~4% steering angle



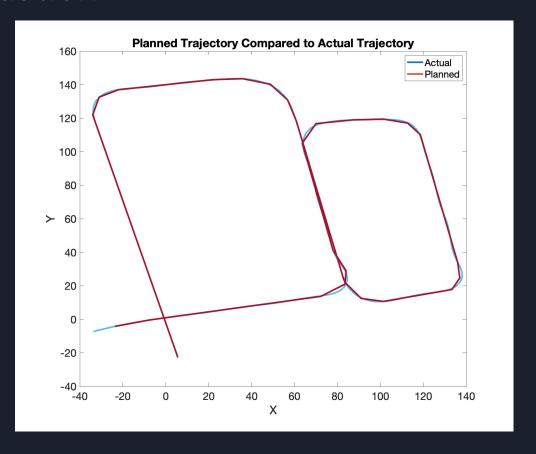
Final Race Results



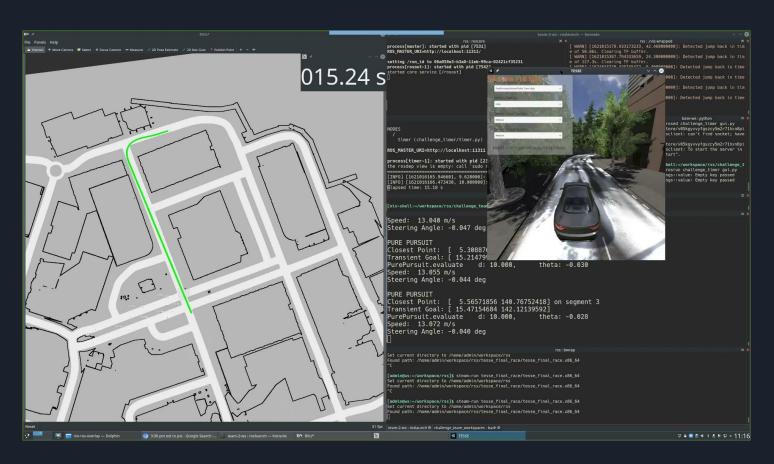




Evaluation

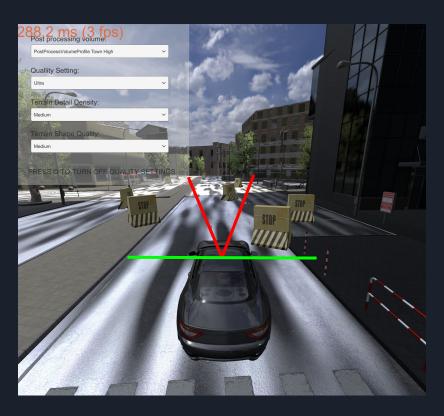


Final Race Video

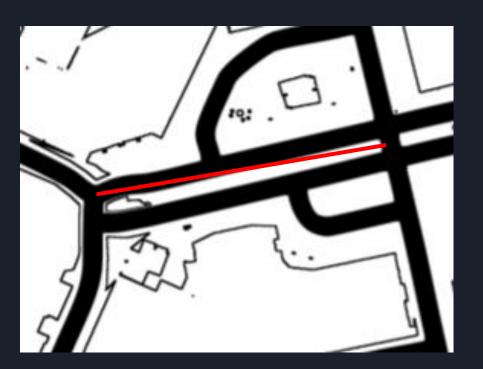


Challenge 2: Obstacle Avoidance

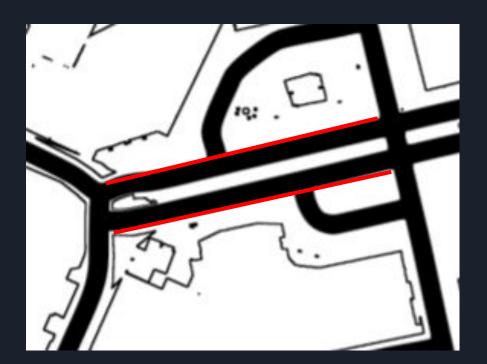
Lidar is used to detect obstacles and make decisions in real time.



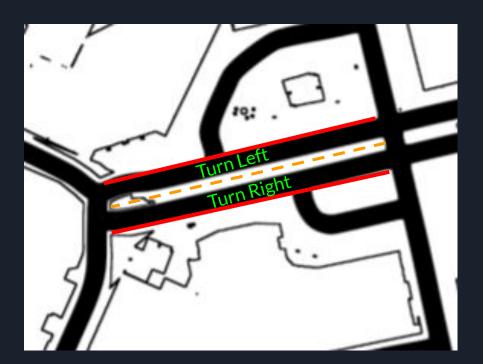
1. Line Following



- 1. Line Following
- 2. Edge Detection



- 1. Line Following
- 2. Edge Detection
- 3. Edge Deflection



- 1. Line Following
- 2. Edge Detection
- 3. Edge Deflection
- 4. Pure Pursuit



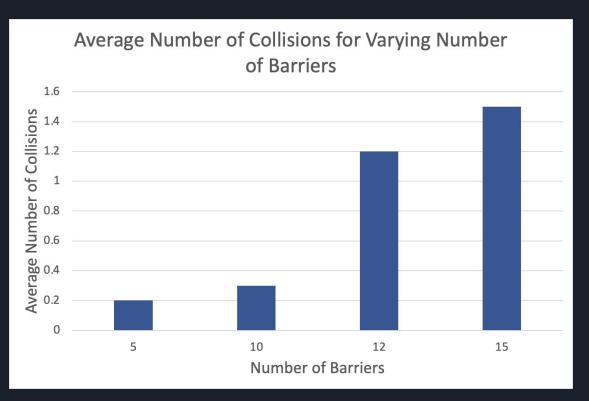
Data Collection Statistics







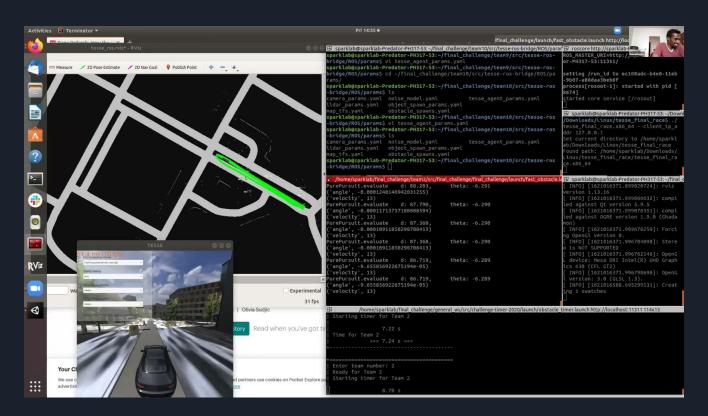
The average number of collisions increased with the number of barriers.



Times from Race Day (Seconds)

7.30 7.24 7.06

Obstacle Avoidance in Action



Technical Takeaways

- Submodules:

 utilized previous labs + augmented code without writing from scratch

- Integration:

- correct implementation + tuned for the challenge-specific requirements
- purely deterministic VS deterministic + probabilistic path
 planning

Communication Takeaways

- Working in pairs / Collaborating in group >> Working alone
 - Github: working in branches, pushing / pulling different from submodules
 - Collaborative trial / error
 - Letting experimental results dictate final code version

Thank You

Questions?