MRSD Project Course

Team I – AIce

Autonomous Zamboni Convoy

Progress Review 7

Team

Rathin Shah

Nick Carcione

Yilin Cai

Jiayi Qiu

Kelvin Shen

Author

Yilin Cai

Step 08, 2022

Contents

1 Individual Progress

Starting from this semester, our team will work on the real Zamboni vehicle and scale up the entire software stack developed last semester to the vehicle. My work in this semester also includes defining the whole software architecture, integrating different algorithm modules in Zamboni with a focus on the planning algorithms.

1.1 Software architecture

Since the Zamboni needs to do drive-by-wire retrofit first, we need a separate platform to do the software development first. We have already tested and validated our software architecture in the RC car platform, which demonstrates the feasibility of it. However, there are still space for improvement:

- The previous autonomy subsystem system is very sensitive to the outliers from the perception subsystem. Once the perception system fail to capture the leader Zamboni or have a outlier with high error, the autonomy system can not recognize it and will make wrong motion decision.
- The accumulated error from the localization and perception will influence the final leader following performance. Since we estimate the leader's trajectory by adding the relative position to the follower's own position, both the error of the localization and perception will be added together. Even we fused the IMU and the wheel encoder for localization, further sensor fusion is required to eliminate errors.
- We rely on a Aruco marker board to detect the leader's pose, which is not an elegant way in real application, especially on the ice rink. We need a board with a large size for detection, which will influence the original operation of the Zamboni vehicle.

So in terms of the software architecture, more information should be used for improvement of both accuracy and robustness.

1.2 Software integration to ATV

We plan to use a Yamaha ATV (All Terrain Vehicles) to start testing our autonomy software and then try to scale up the the Zamboni. Fig. 1 shows the interface of our previous algorithm flow to the ATV software.

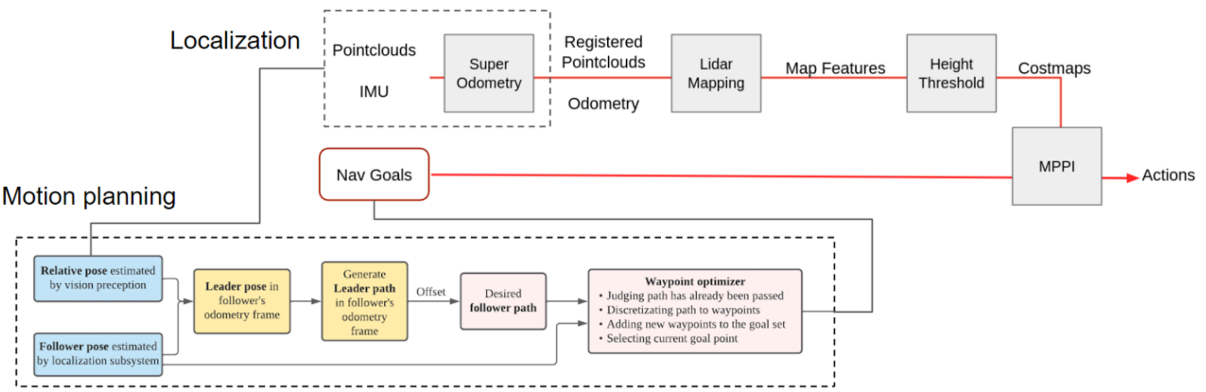

Figure 1: Integration of previous autonomy algorithm to the ATV software.

In the motion planning part, the leader's path is recovered by adding perception result to the follower's own position. After leader path recovery, we add a lateral offset to the path to get the desired path for the follower, because we want the two Zamboni going along different paths. Then the waypoints optimizer includes waypoints updater and waypoints publisher. It will select waypoints for follower tracking based on the follower's pose and velocity. We then update the current targeting waypoints to the low level controller by finding the closest waypoint in the path to follower's current pose. Then we extract the local path with constant waypoint numbers for controller to track.

The ATV has already provided us a localization module with super odometry, which will improve our localization subsystem. Also by passing waypoint as a navigation goal to the AVT, it will be able to follow the leader's trajectory.

2 Challenges

The major challenge lies in the uncertainty in the physical vehicle. Once the Zamboni or the ATV is able to be used, we can test our software as soon as possible.

3 Teamwork

Kelvin Shen is in charge of perception and recognition. His progress includes:

- Refreshed memory of our codebase as well as ROS usage.
- Reviewed ATV autonomy stack and modified our functional architecture accordingly.

Rathin Shah is in charge of controlling the real RC car. His contribution includes:

- Finalized Functional Requirements
- Studied Motor Controller for CANbus communication for throttle control.
- Studied electric throttle for voltage range
- Researched different dbw solutions for steering.

Nick Carcione is in charge of the DBW hardware. His contribution includes:

- Updated performance requirements.
- Brainstormed potential dbw steering design.
- Reviewed ATV autonomy stack.

Yilin Cai is in charge of the planning for the robot and software integration. His contribution includes:

- Learnt and review the ATV autonomy stack
- Conception of transferring and fusion of previous autonomy codebase to ATV

Jiayi Qiu is in charge of the leader's estimation and control algorithm development. Her contribution includes:

- Worked on the collaboration with Isuzu, updated potential dbw steering and braking designs
- Reviewed ATV autonomy stack and installed dependencies to build the repository.

4 Plans

The next step includes software interface, further testing and new fusion algorithm development. As the defect described above, we also plan to use better controller rather than the pure pursuit. And we need to look at the real vehicle and work on it quickly as soon as possible.