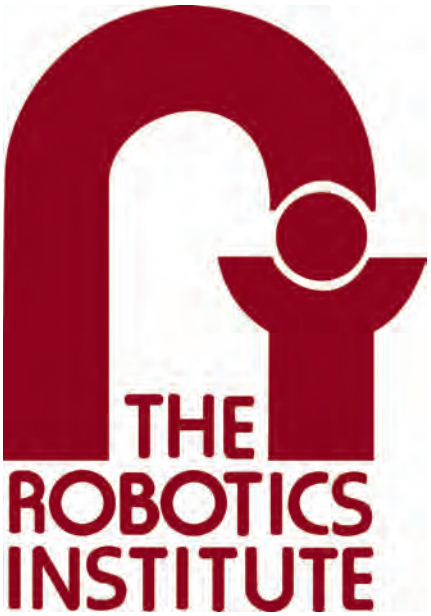

MRSD Project Course

Team I – AIce

Autonomous Zamboni Convoy

Progress Review 1

Team

Rathin Shah

Nick Carcione

Yilin Cai

Jiayi Qiu

Kelvin Shen

Author

Yilin Cai

Feb 17, 2022

Contents

1 Individual Progress

My responsibility for the Autonomous Zamboni Convoy project mainly focus on setting up and maintenance of the simulation environment. My work also includes defining the whole software architecture, integrating different algorithm modules, and providing algorithm interface from simulation environment.

1.1 Simulation setup

First, I developed the URDF file for the Zamboni vehicle, which had an Ackermann steering mechanism. To better organize the model description, I used XACRO file to build the vehicle model.or better visualization, I built a detailed mesh file for the Zamboni, which was exported from the Solidworks CAD model provided by the sponsor. However, the CAD model has over 1,000 parts describing all the details of a car. I combined all the parts into only five: the four wheels and a car body, and exported them as mesh files. Due to the complexity of the original mesh file, gazebo will get stuck when loading the vehicle. Thus I used MeshLab to simplify the faces and vectors on of the mesh file, deleting unnecessary detail while keeping the major outer appearance. For better visualisation, I used Blender to change the scale, center position, and more importantly, the color of each face.

In URDF file, to align the frame of them and modify then to finally make the vehicle looks more realistic was very time-consuming. The whole setting up process is shown in Fig. 1. For the follower Zamboni, in addition to the vehicle model itself, I added IMU, LiDAR and camera sensor to it using Gazebo plugins. The controller interface in gazebo, including the rear wheel velocity and front wheel steering angle controller was also setup. Moreover, I developed the keyboard controller sending velocity and steering angle to the two vehicles individually. Now

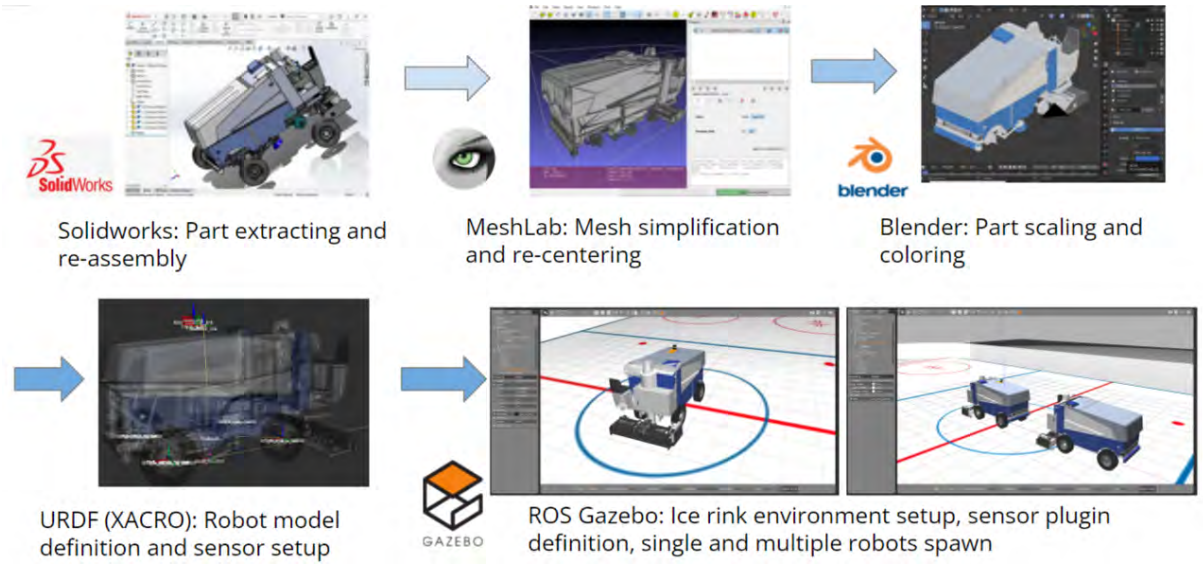

Figure 1: Zamboni simulation model setup process.

1.2 Multi-vehicles Spawn

To define two different robots in simulation, although the URDF definition and configuration files for the two robots are the same or similar, the ROS topic should be different in order to

distinguish and identify them. Thus in launch file, different namespaces for the vehicles should be defined. Then we can send separate topic to the two vehicles individually.

ROS *tf* information is quite important for robot simulation, especially in multi-vehicles simulation. The links in each vehicle are linked to there own base frame, however, the relative pose of the two vehicles also need a 'abstract' frame. So for the leader vehicle, I got the robot state from gazebo and published its abstract pose and velocity, which could be the ground truth. And I used the *tf* tranformation broadcast to define a map frame for it. For the follower, I defined the odomtry first and link the map to the odomtry, which was then connected to all vehicle links. Figure 2 and Figure 3

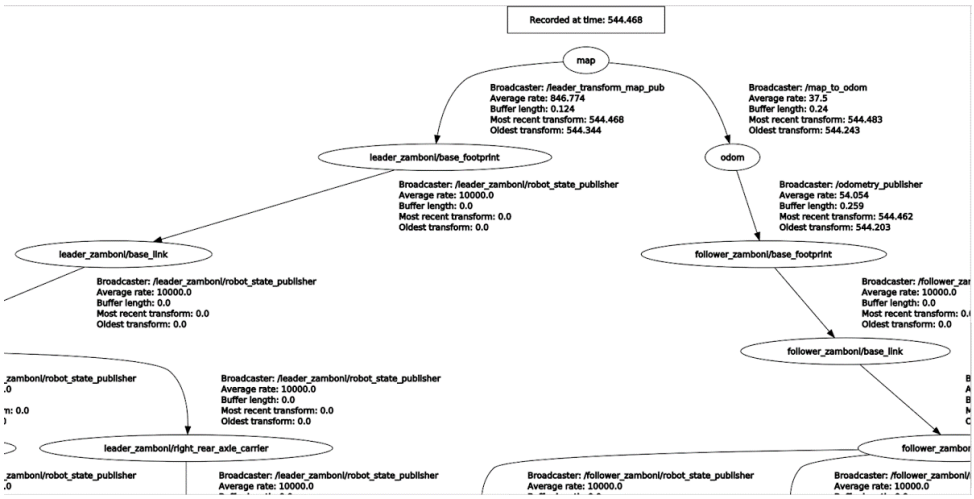

Figure 2: *tf_tree* of the linking the two vehicle to the map frame.

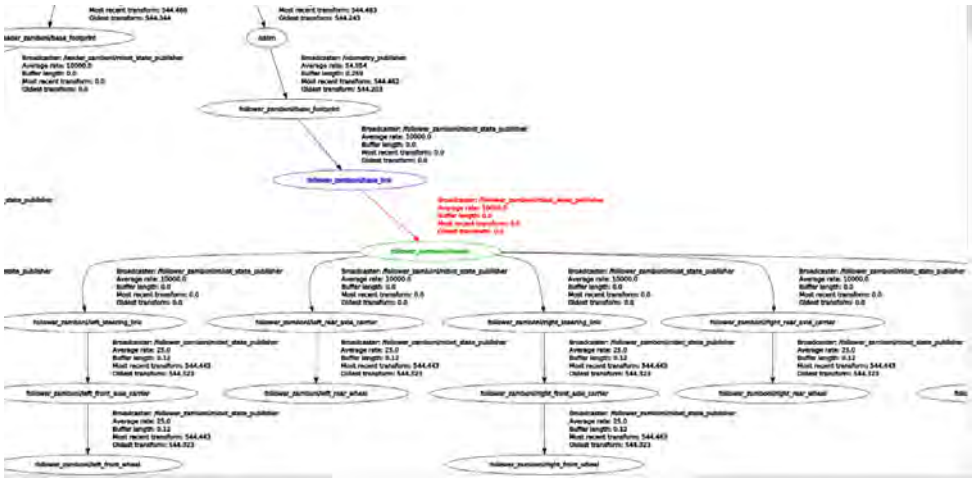

Figure 3: *tf_tree* of one robot.

1.3 Follower localization

To realize the following the leader motion, we define the path for the follower in its own origin frame by adding leader's relative pose it sensed to it own location. Here I developed an wheel odometry of the follower Zamboni. Also I integrated the *ros_localization* package into the simulation, which can use the Extended Kalman Filter to fuse the IMU information and odometry information. The result is shown in Figure 4, where the odometry path is recorded in Rviz.

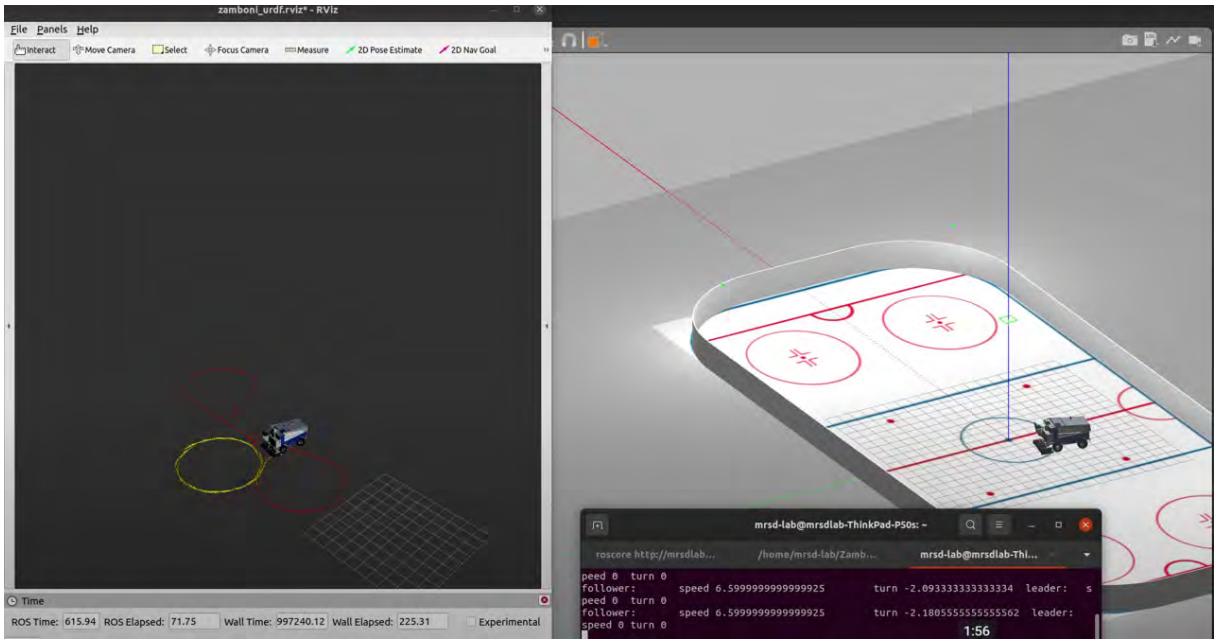

Figure 4: One vehicle localization.

2 Challenges

The first major challenge lies in spawning multiple robots in the gazebo simulation. I found the tf frames to be effectively an entire parallel namespaces where the "tf_prefix" was even more challenging to handle in a scalable way. Normally wrapping <group ns="$(arg robot_id)"> around the launch file of the single robot will allow to run multiple instances of a node in namespaces and put topics and parameters into a nested namespace. Declaring a nested tf_prefix parameter will prepend a namespace to the coordinate frames being published by each robot_state_publisher node based on the robot_description. In conclusion, to define separate links in tf_tree, only the namespace is not enough, a "tf_prefix" defined in launch file is required.

The second challenge is about the robot localization. Since the odometry is based on wheel velocity, wheel slippery will definitely cause error to the localization using odometry. However, I came across the situation that when the robot hit the wall, although it is not moving, the wheels are still rotating. In this situation, the odometry will fail. To solve this situation, I planned to do a collision check that the vehicle will receive a zero-velocity command to really stop. Also I will check the physical parameter setting in both gazebo and URDF to solve this problem.

3 Teamwork

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

- Integrated RealSense D435i into Zamboni model in Gazebo.
- Estimated pose of the leader Zamboni using ArUco board and the RGB image
- Extracted depth values in the Depth image after cropping out the region of interest based on the marker detection.

Rathin Shah is in charge of the controller development. His contribution includes:

- Progressed on the Localization of the robot and validated its performance.

- Integrating the Pure Pursuit Controller developed in Simulink with the simulation setup in gazebo.
- Made the schematic for Power Distribution Board-PCB .

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

- Got familiarized with current Zamboni architecture and identified hardware components necessary for drive-by-wire conversion and began looking into their availability.
- Acquired the backup RC car platform.
- Created the management style presentation.

Yilin Cai is in charge of the simulation setup. His contribution includes:

- Spawned multi-robots in Gazebo and command the two Zamboni moving individually by keyboard teleoperation.
- Develop wheel odometry for follower Zamboni.
- Command vehicle motion command with keyboard teleoperation.
- Integrate robot localization package fusing odometry and imu for follower's pose estimation.

Jiayi Qiu is in charge of the simulation environment setup and leader's estimation. Her contribution includes:

- Solved ROS version conflict when launching Zamboni URDF model.
- Familiarized with ROS navigation including teb_local_planner plugins, move_base package.
- Loaded predefined waypoints to Zamboni simulation and visualized the path in Rviz.

4 Plans

The next step of my MRSD project works will still focus on the simulation part. I will work towards more accurate Localization and estimation of follower Zamboni with the sensor model available. I will work on the camera and LiDAR data integration. In addition, I will continue refinement of simulation details, like physical parameter in URDF file and gazebo controller plugins. I will also modify the mesh file to equip the leader Zamboni with a ArUco marker figure at the back of it, which will create the interface for detection. Moreover, I will use the detected pose information to generate the waypoint, and then use *ROS_Navigation* package to track the waypoints.