WPI

Optimal Path Planning for Autonomous Collaborative Delivery Robots with Dynamic Obstacle Avoidance

RBE 550 Final Project Presentation

Akshay Laddha
Anagha Ramaswamy
Kunal Nandanwar

PROJECT SCOPE

Goals

Goals at the start of the semester:

- Environment mapping on Carla and Gazebo
- Implementation of RRT* motion planning algorithm
- Obstacle avoidance (static + dynamic)
- Implementation on swarm of robots

Revised Goals:

- Environment mapping on Gazebo and PyGame
- Implementation of RRT motion planning algorithm and its variants
- Obstacle avoidance (static on Gazebo + dynamic on PyGame)
- Implementation on swarm of robots

TOOLS

Tools

- PyGame
- Gazebo
- MoveIt
- Rviz

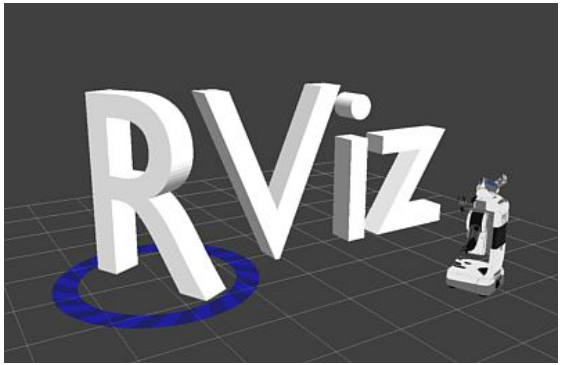
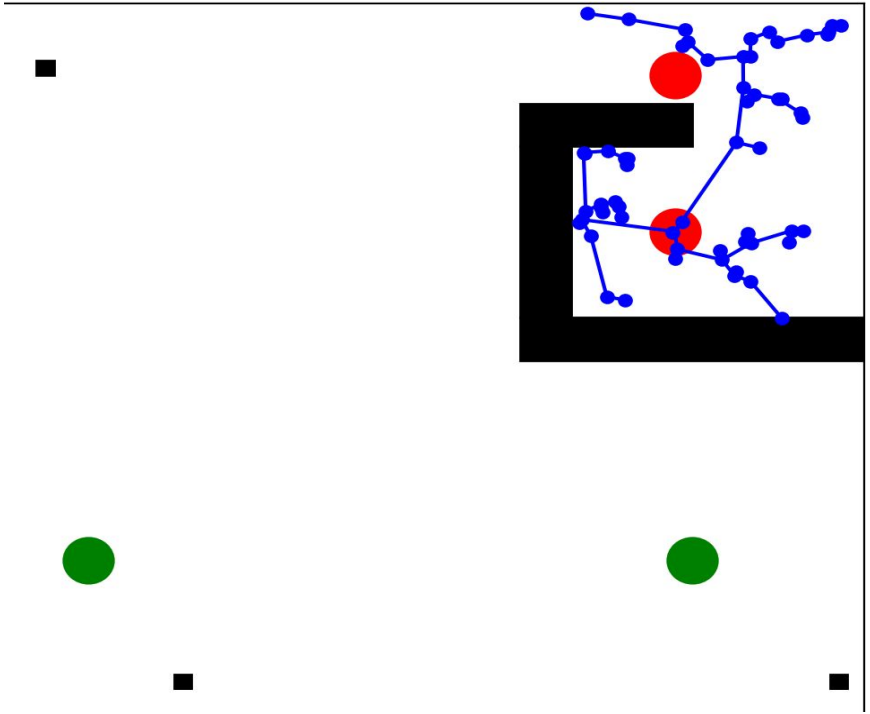
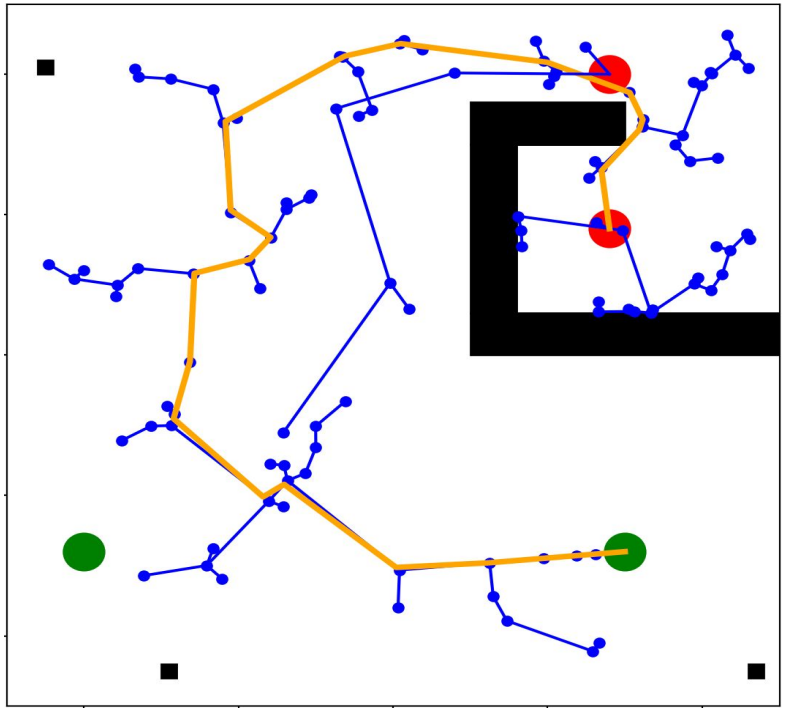


Image Courtesy: https://www.pygame.org/news
https://www.coppeliarobotics.com/
http://gazebosim.org/
https://carla.org/

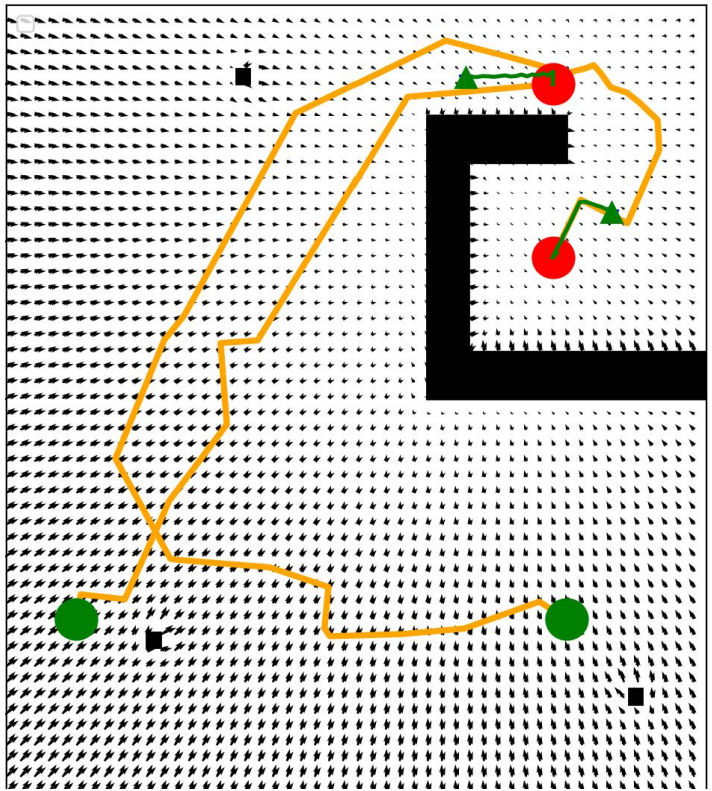
PyGame

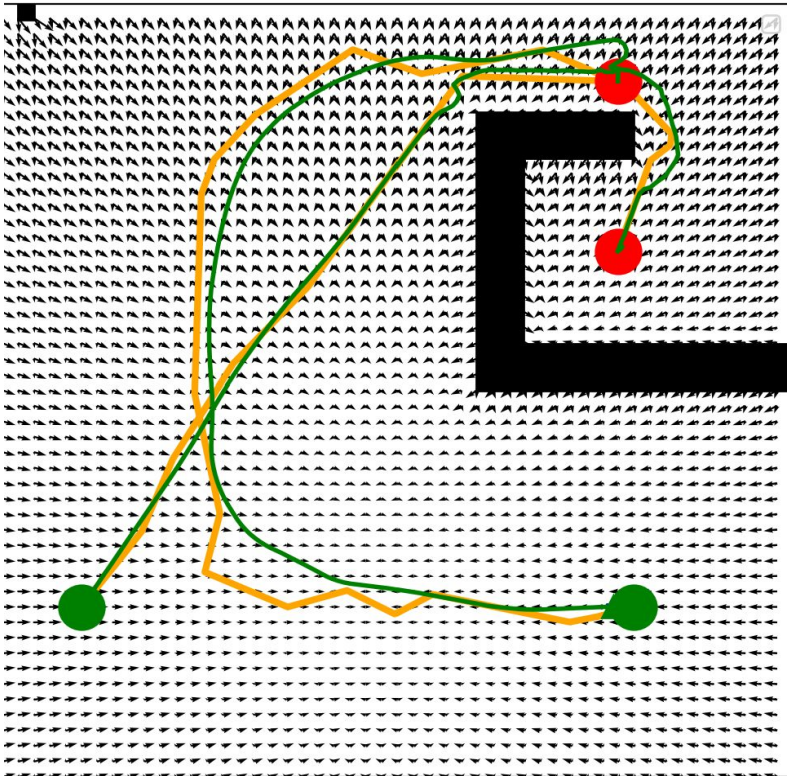
PyGame





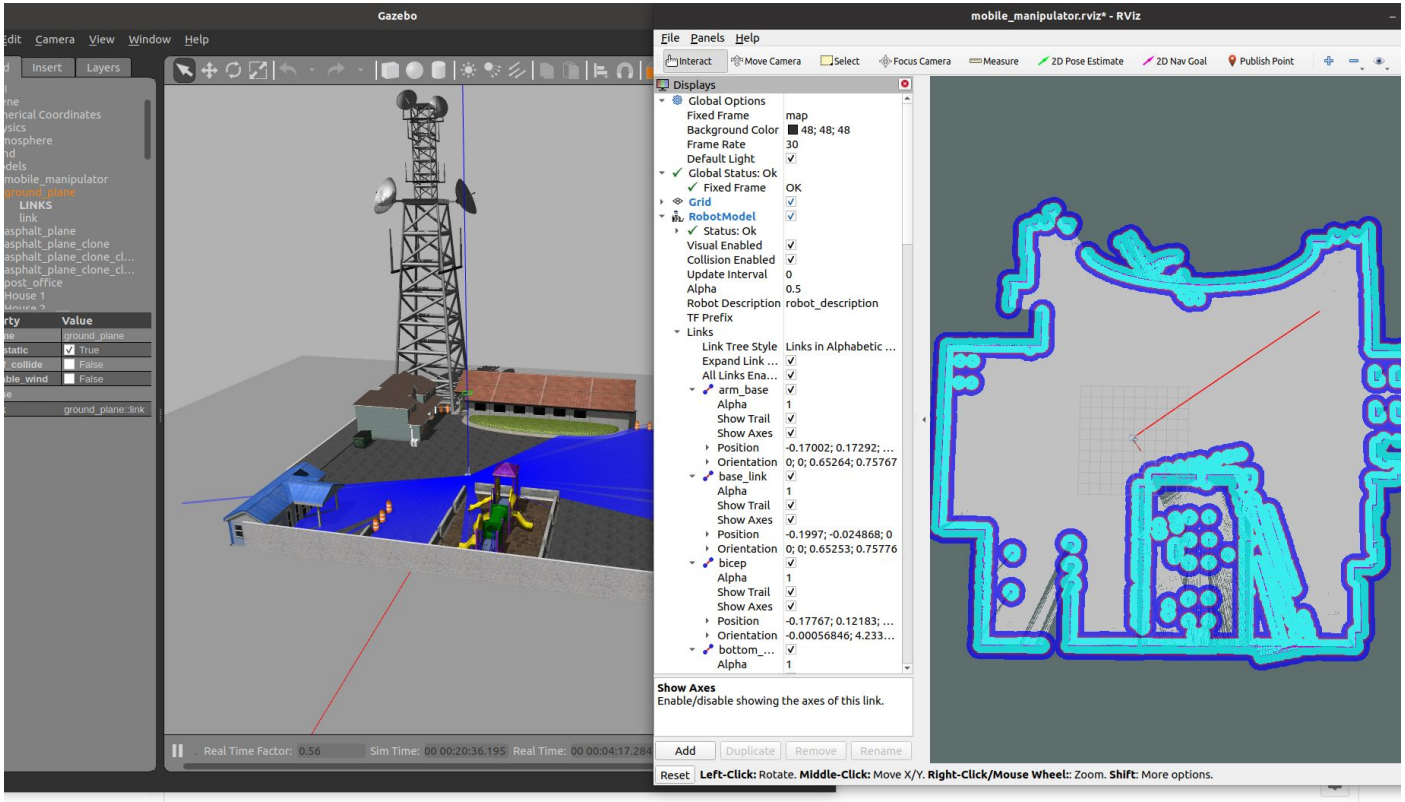
PyGame



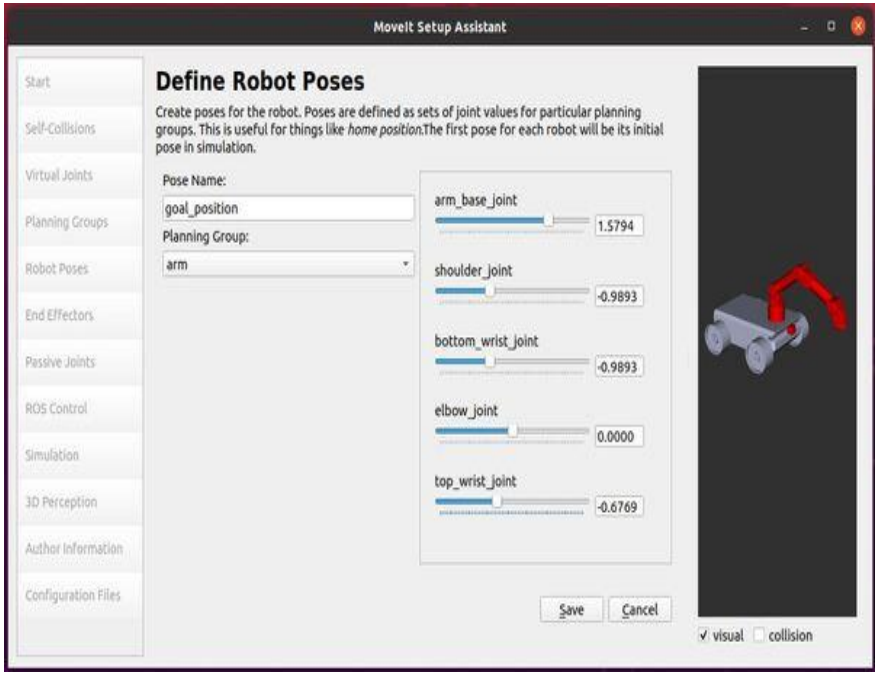


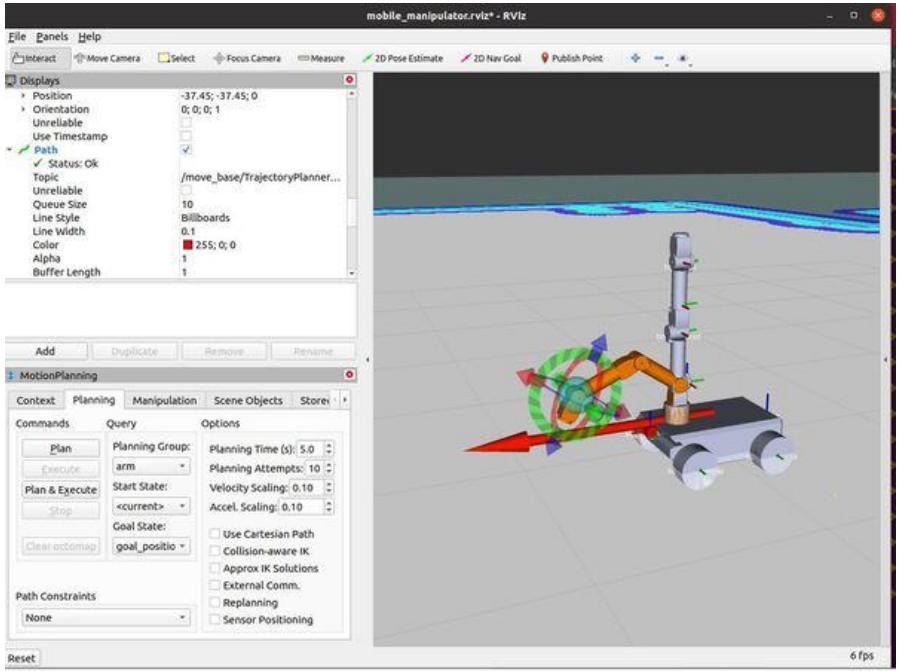
GAZEBO

Gazebo - Single Robot Delivery

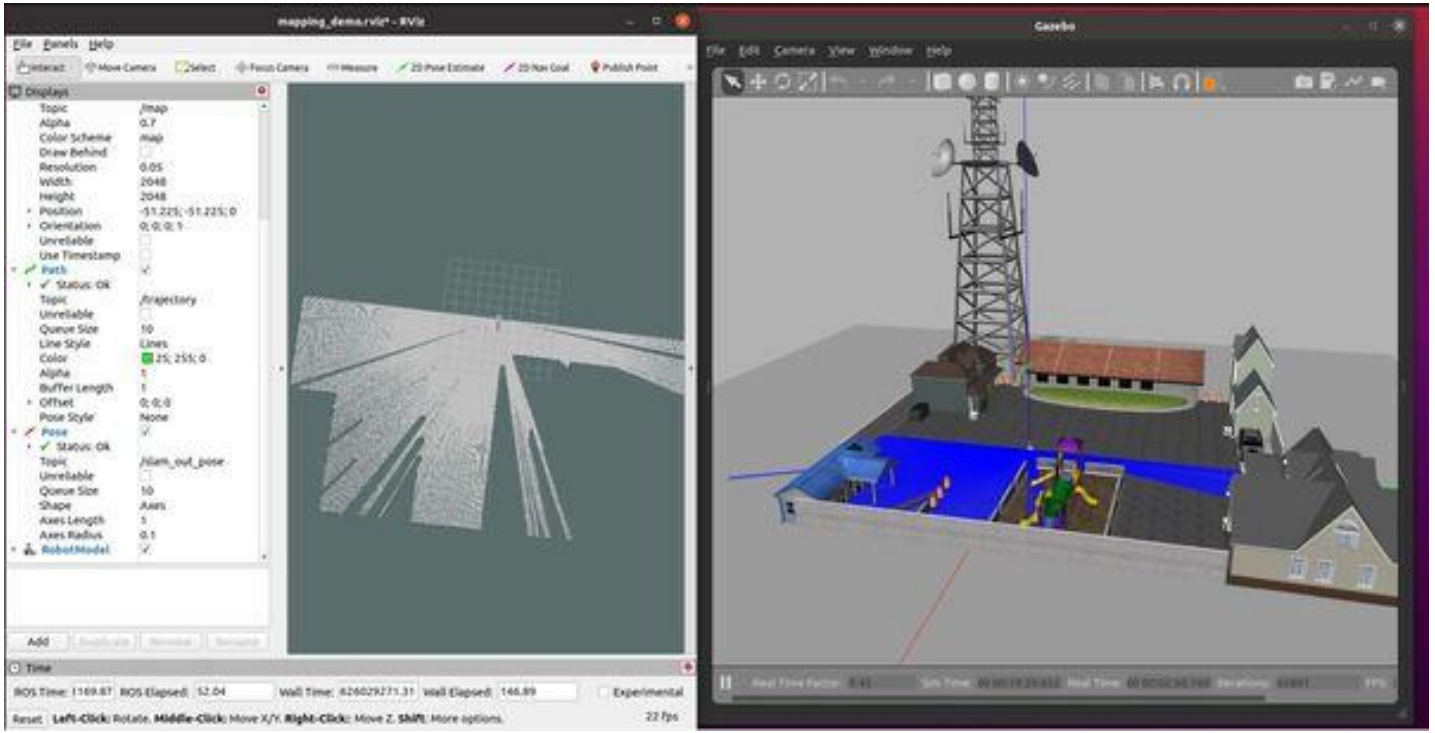


Single Robot Delivery - Controlling Robotic Arm using MoveIt & ROS

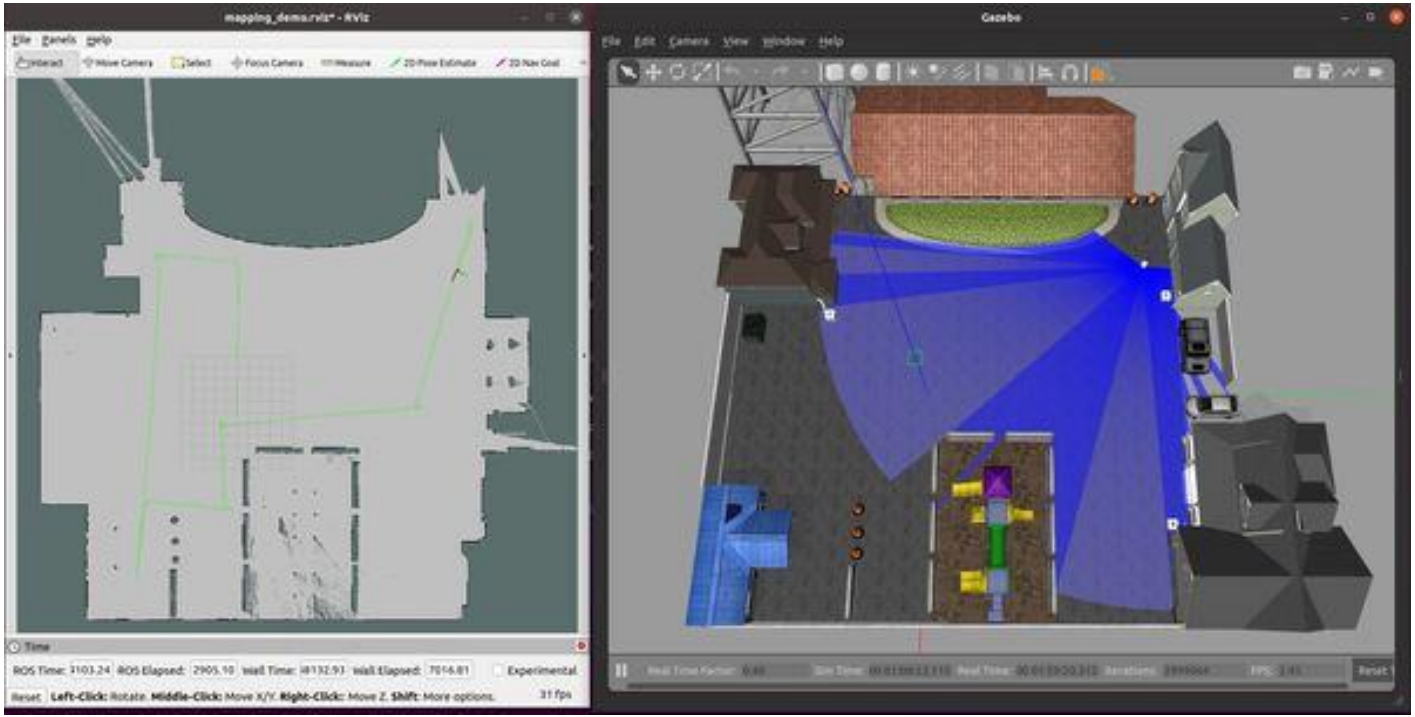




Single Robot Delivery - SLAM using Hector-SLAM

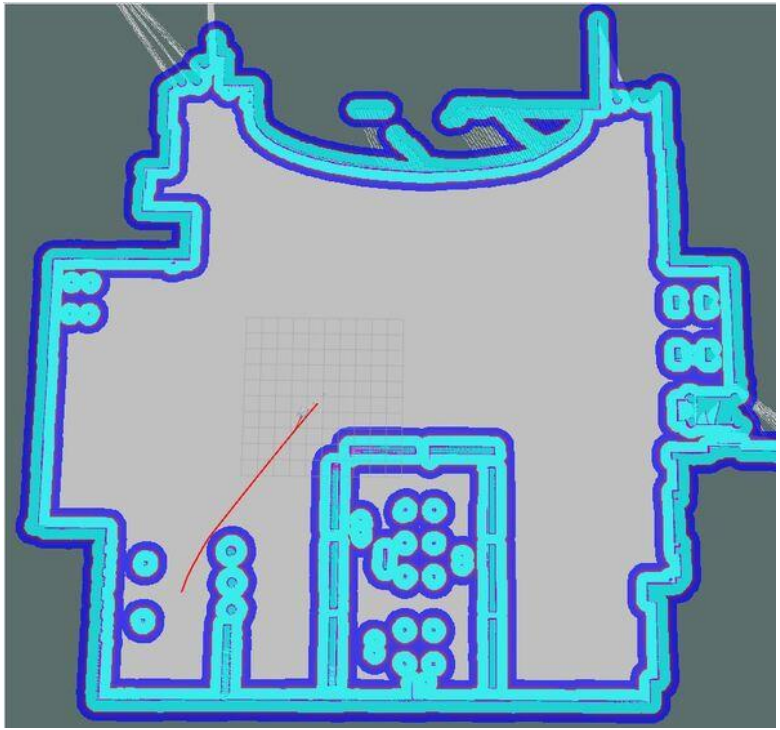


Single Robot Delivery - SLAM (contd.)

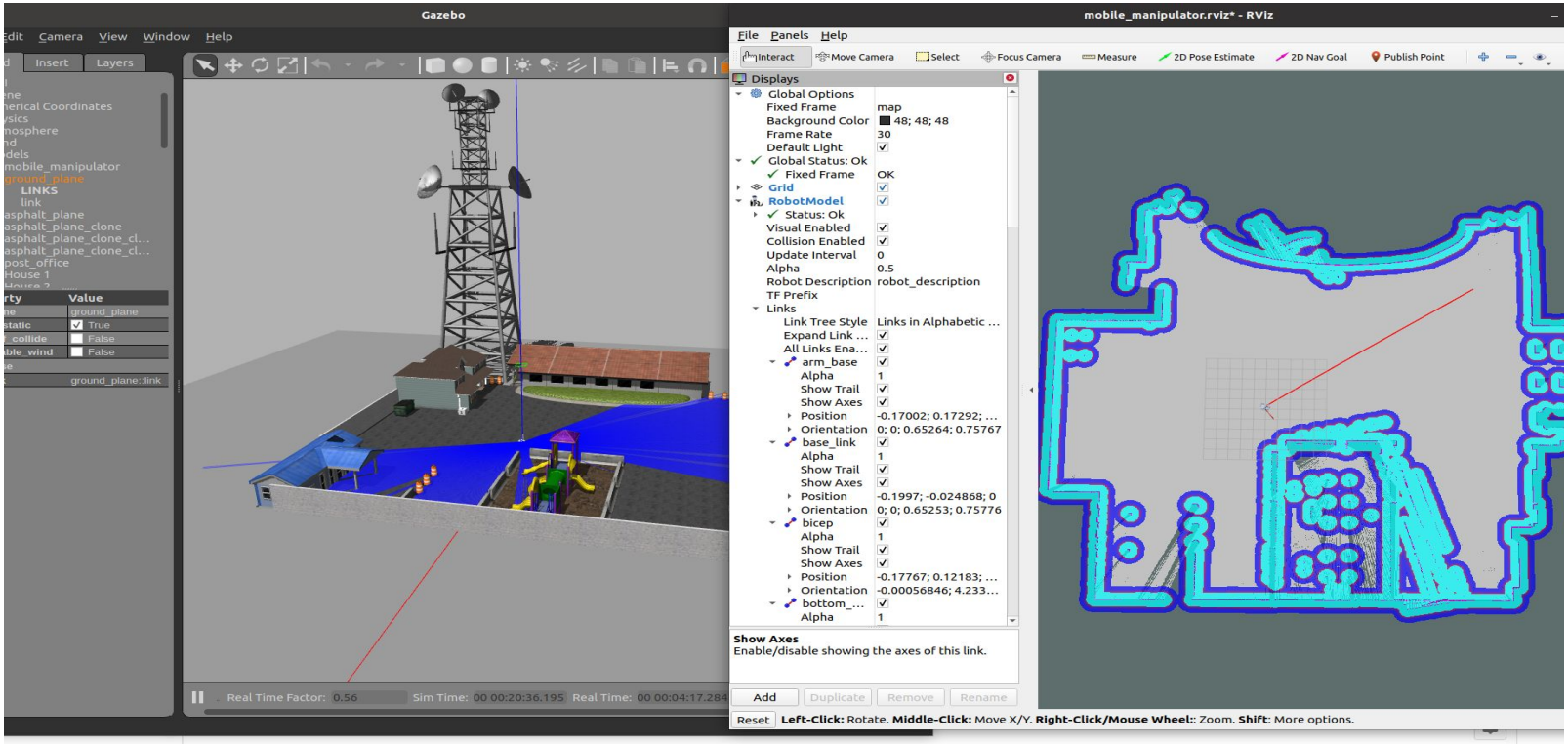


Single Robot Delivery - Sending robot to goal location using ROS

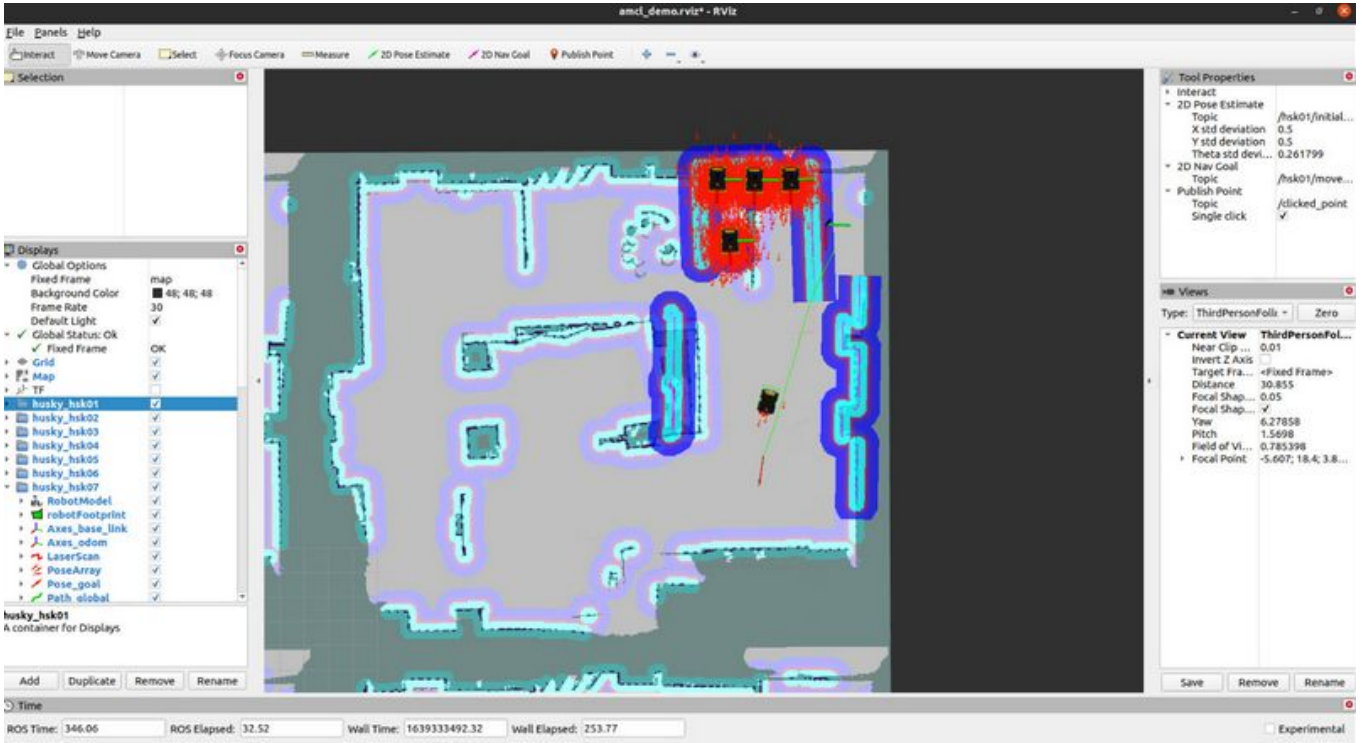
```
akshay@akshay-Nitro-AN515-42: ~/catkin_ws
not mix paths from different distributions.
ROS_DISTRO was set to 'noetic' before. Please make sure that the environment doe
s not mix paths from different distributions.
akshay@akshay-Nitro-AN515-42:~$ source /opt/ros/noetic/setup.bash
ROS_DISTRO was set to 'foxy' before. Please make sure that the environment does
not mix paths from different distributions.
akshay@akshay-Nitro-AN515-42:~$ source /opt/ros/noetic/setup.bash
akshay@akshay-Nitro-AN515-42:~$ cd ~/catkin_ws
akshay@akshay-Nitro-AN515-42:~/catkin_ws$ source devel/setup.bash
akshay@akshay-Nitro-AN515-42:~/catkin_ws$ rosrun mobile_manipulator send_goals

Where do you want the robot to go?

1 = House 1
2 = House 2
3 = House 3
4 = Post Office

Enter a number: 4

Goal Location: Post Office

[ INFO] [1650842877.484692167, 1152.263000000]: Sending goal
```



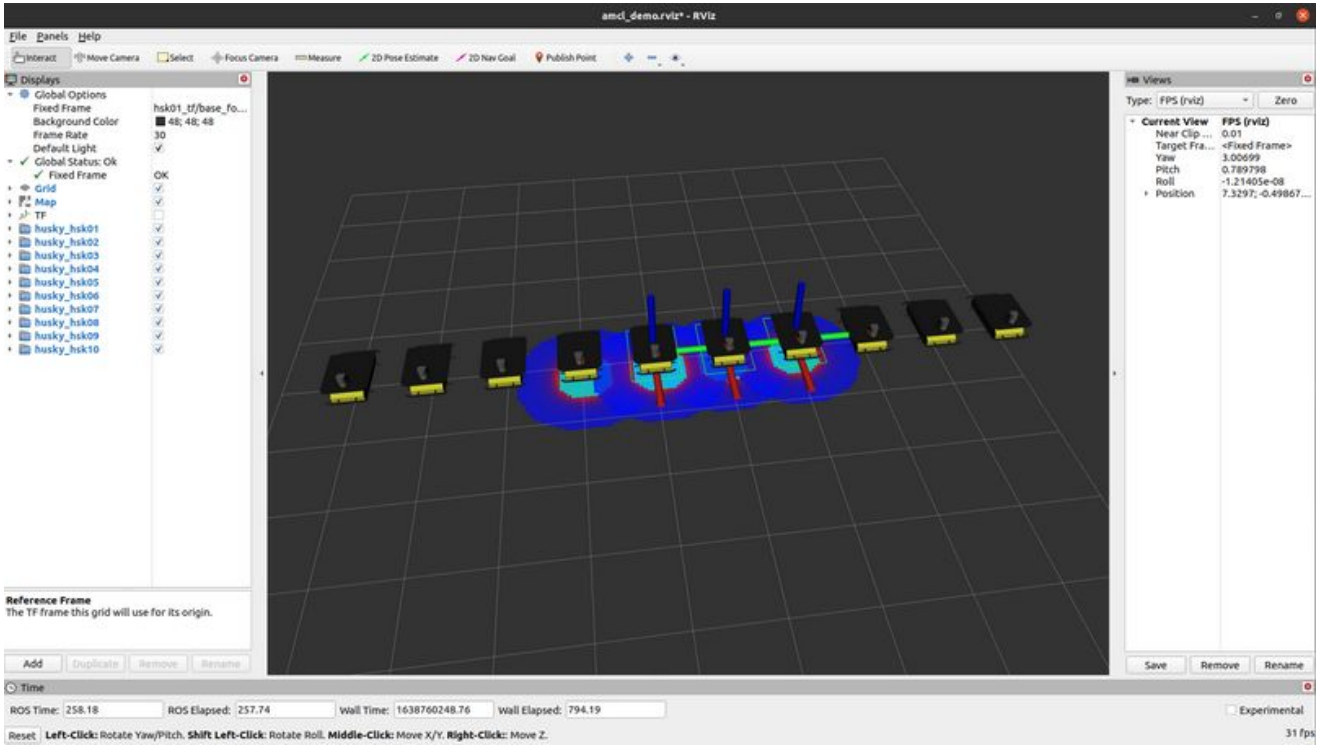
Single Robot Delivery - Sending robot to goal location using Rviz



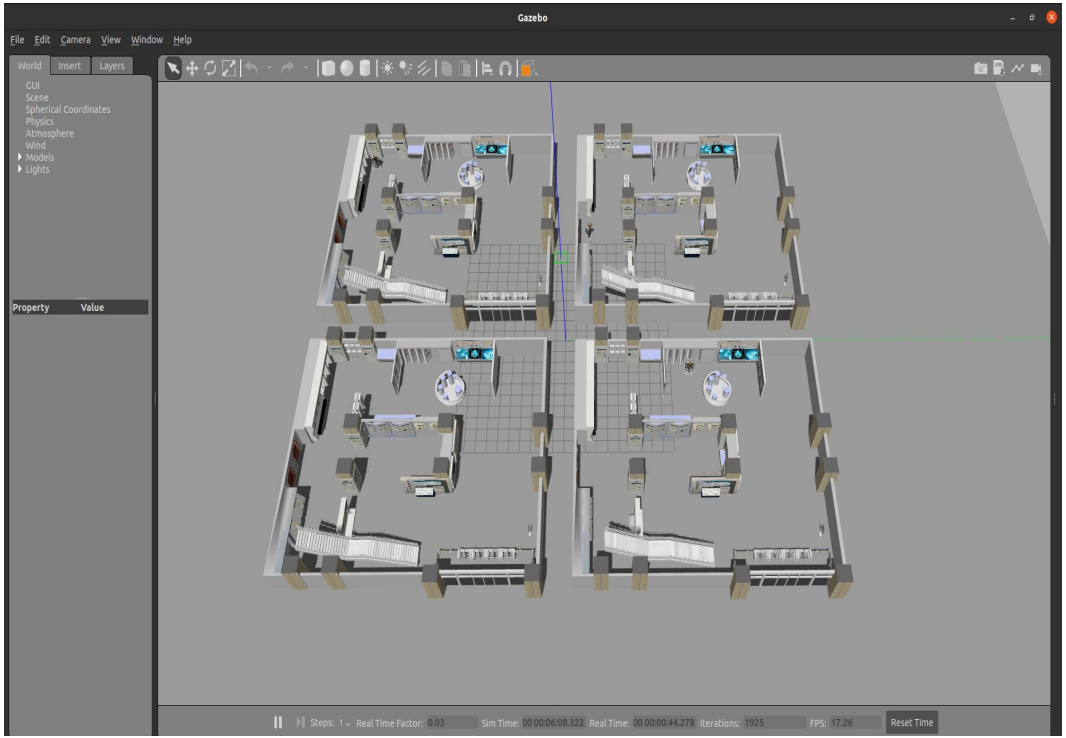
Gazebo - Multi Robot Delivery



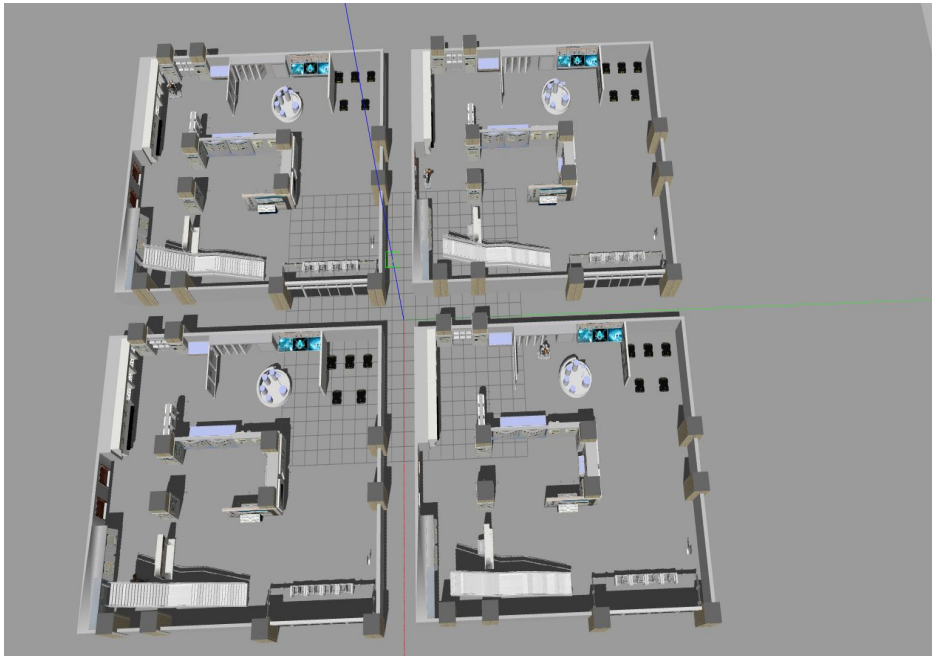
Multi Robot Delivery - Setting up the Robots

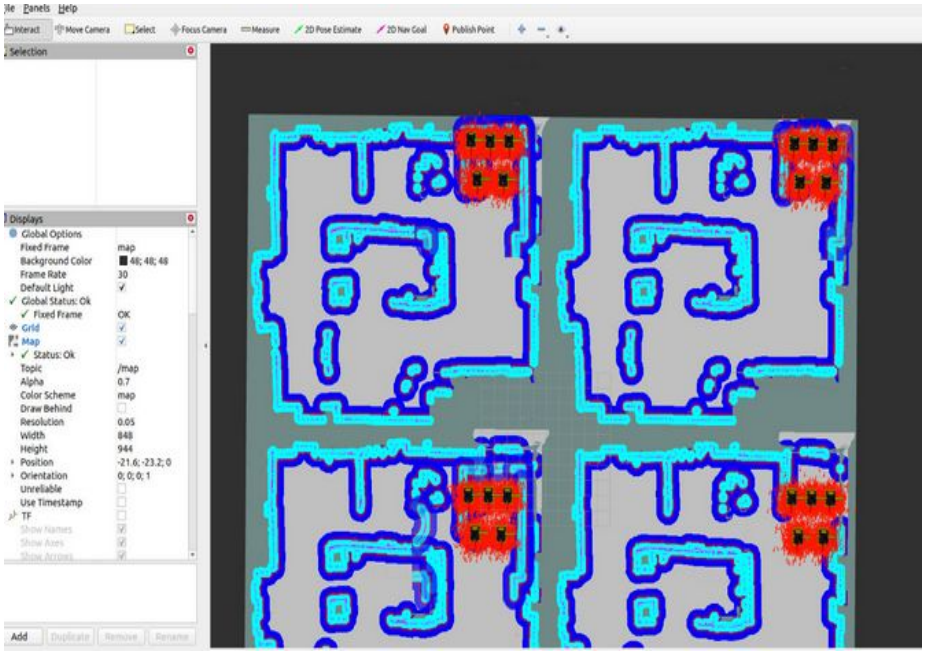


Multi Robot Delivery - SLAM using Gmapping



Multi Robot Delivery - Final Robot visualization map





Multi Robot Delivery - Robot heading to goal location

(Simulation Video, attached outside)

Issues

Because of the significant padding in the cost map generated, the robots are unable to discover paths due to restricted pathways.

Using smaller robots or designing alternate pathways can help reduce this.

Risks & Mitigation

Risk 1:

Error in the Path Planning Algorithm owing to unknown limitations, which might result in a delivery delay or failure.

Mitigation:

Can be improved by feeding path planning data to supervised learning ML techniques and training the path planner to forecast obstacles/possible unknown limitations.

Risks & Mitigation

Risk 2:

If ROS Master is unexpectedly shut down, all accessible agents/Robots may be shut down as well.

Mitigation:

This risk can be reduced by either transferring the entire project to ROS2 or integrating a ROS1 bridge with ROS2.

Risks & Mitigation

Risk 3:

One of the path planning sensors may malfunction, resulting in erratic behavior.

Mitigation:

If this happens, the robot will be forced to override the autonomous navigation signals and will be controlled manually by a human operator.

Conclusion

Successfully implemented a single and multi robot delivery system in 2D using PyGame and in 3D using Gazebo, MoveIt and RVIZ

Thank you!
Any Suggestions?

By:
Akshay Laddha
Anagha Ramaswamy
Kunal Nandanwar

Image Courtesy: https://grabcad.com/library/autonomous-delivery-robot-1