

Linking ROS to Gazebo

Kenzie King

Sprint 3 Documentation of Work

Objective of Research

The objective of this investigation was to learn how to programmatically link ROS compatible code to the Gazebo simulation environment. This would allow us to ensure our controls code has a method of communication with our environment, without which, there would effectively be no simulation.

Findings of Research

ROS provides a wrapper around Gazebo that allows for simulation of a robot using ROS messages in the same manner that would be used in the real world with proper sensor data.

Since the default Gazebo sensor models do not include ROS integration by default, we will have to write our own publisher node to output the data from the simulated sensor to our ROS-based processing code.

ROS requires robot models to be in URDF format as opposed to SVG which is the preferred format for Gazebo. This means that within our URDF model, we will have to include various elements of extra information manually. Without this, we would have a visual model, but the links and joints would not have any motion since Gazebo would not know what to do with them.

One very interesting find of this research topic is that dynamic reconfigure (RQT) can be used with Gazebo. RQT is a ROS tool that allows a user to tune certain parameters on the fly to see the results immediately.

Research Conclusion

There are some unexpected struggles associated with linking ROS and Gazebo. Since ROS is one of the main uses for Gazebo, I had assumed that it was more compatible. There will need to be many adjustments to how we build our vehicle models compared to what was expected as well as how our sensors will function. One positive note is that with this integration, we can use dynamic reconfigure to tune parameters that we see fit on the fly.