Link to bootcamp repo: `https://github.com/UTSARobotics/ros1_bootcamp`

1 Introduction

Today's activity will add to Day 3's activity. We'll also go over how to use bag files!

2 Message types, Bags, and Plots

2.1 Service Server

Use the node you implemented in Exercise 3 and add a service server that can start/stop the robot. This functionality could be used as an emergency stop.

1. Implement a service server that you can start and stop the robot. Use the `std_srvs/SetBool` service type for this task. Load the service name from the parameter server.
2. Run the simulation and call the service you have implemented from the terminal using `rosservice call` to start and stop the robot. You need to implement the start/stop request handling logic such that you can call the service multiple times without restarting the simulation.

2.2 Bags

1. Download the provided rosbag `smb_navigation.bag`: link here!
2. Write a launch file that starts an `ekf_localization_node` subscribing to the topics provided in the rosbag file. Load the parameters from the same config file as it is done by the simulation (Tip: The config file `localization.yaml` can be found with `roscd smb_control/config`).
3. Use `rqt_mulitplot` to plot the path of the recorded robot in the `x`/`y`-plane.
   - Tip: Remember to set the parameter `/use_sim_time` to `true`.
   - Tip: Play the bag-file with: `rosbag play mydata.bag -clock` which also publishes the time of the recorded data.
4. Visualize the motion of smb by using TF markers in RViz. Add a `robot_state_publisher` node to your launch file and load the smb robot description to the parameter server. Now you can visualize the smb model in RViz. (Tip: Use the `control.launch` file from the `smb_control` package as reference). Note: The wheels jump behind the robot body a bit, but that is okay.
5. The rosbag also contains laser data from a Velodyne LiDAR. Visualize the point cloud in RViz. It should be moving with the robot.