instruction_nav_stack_real_drone_part2

Prajwal Thakur

2022-11-14

Overview

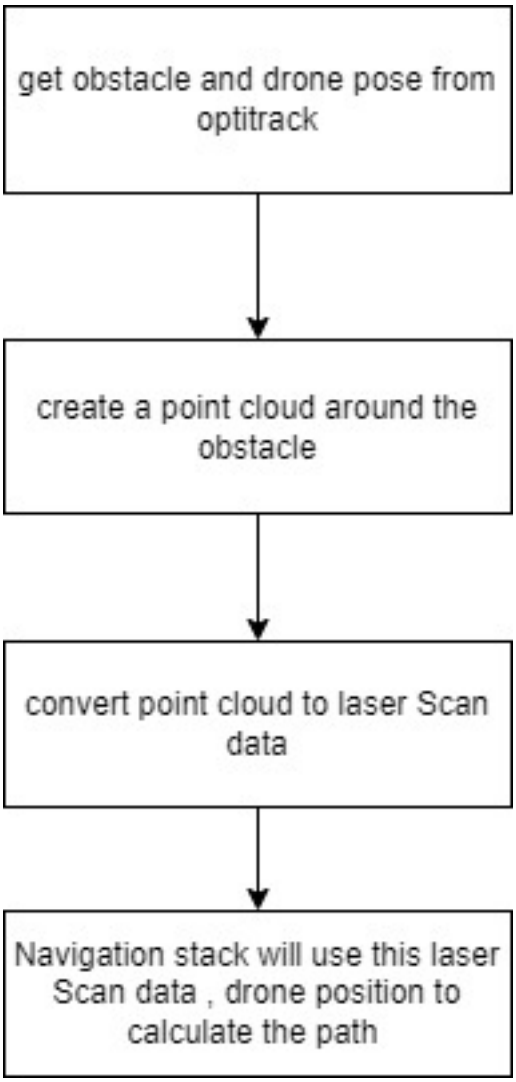


Figure 1: nav_real

Steps

- All the neccesry files are in lab_10_part2_helper.zip
- do the all the following steps in the worsksapce that you have created in lab10
- copy launch files from the lab_10_part2_helper.zip folder in your workspace
- Create a new package having dependencies having following dependencies:
  - rospy
  - sensor_msgs
  - std_msgs
  - message_generation
- copy the the scripts provided in lab_10_part2_helper in above package also copy the point_cloud_circular_.py from lab9
- make the necessary changes in the cmake list
- launch_real_world.launch will be the main launch file
- notice the different launch files being used and difference from the lab9 main lanch file , here we are not calling any gazebo ,
- ref figure 1 and notice the steps that need to be followed
- complete the to do in launch_real_world.launch
- to get the pose obstacle from optitrack we will use
  - get_position_obs_from_optitrack.launch , to get get the obstacle pose from optitrack
  - complete the todo in above launch file and corresponding scripts

TO get pose of the drone

overview

steps

- use get_position_drone_from_optitrack.launch file , to get drone pose from optitrack
- In get_pose_from_optitrack.py , we are subscribing the pose of the drone from the optitrack covert this messages to odometery messgaes and also we create a new base_footprint frame . (We have to create this base_footprint ourselves.)
- refer figure 2

covert obstacle center to laser scan

- copy and USe point_cloud_circular_.py from lab9 to subscribe the /obs_pose_list and creating a point cloud
- copy and use point_cloud_to_laser.launch from lab9 to convert point cloud to laser scan

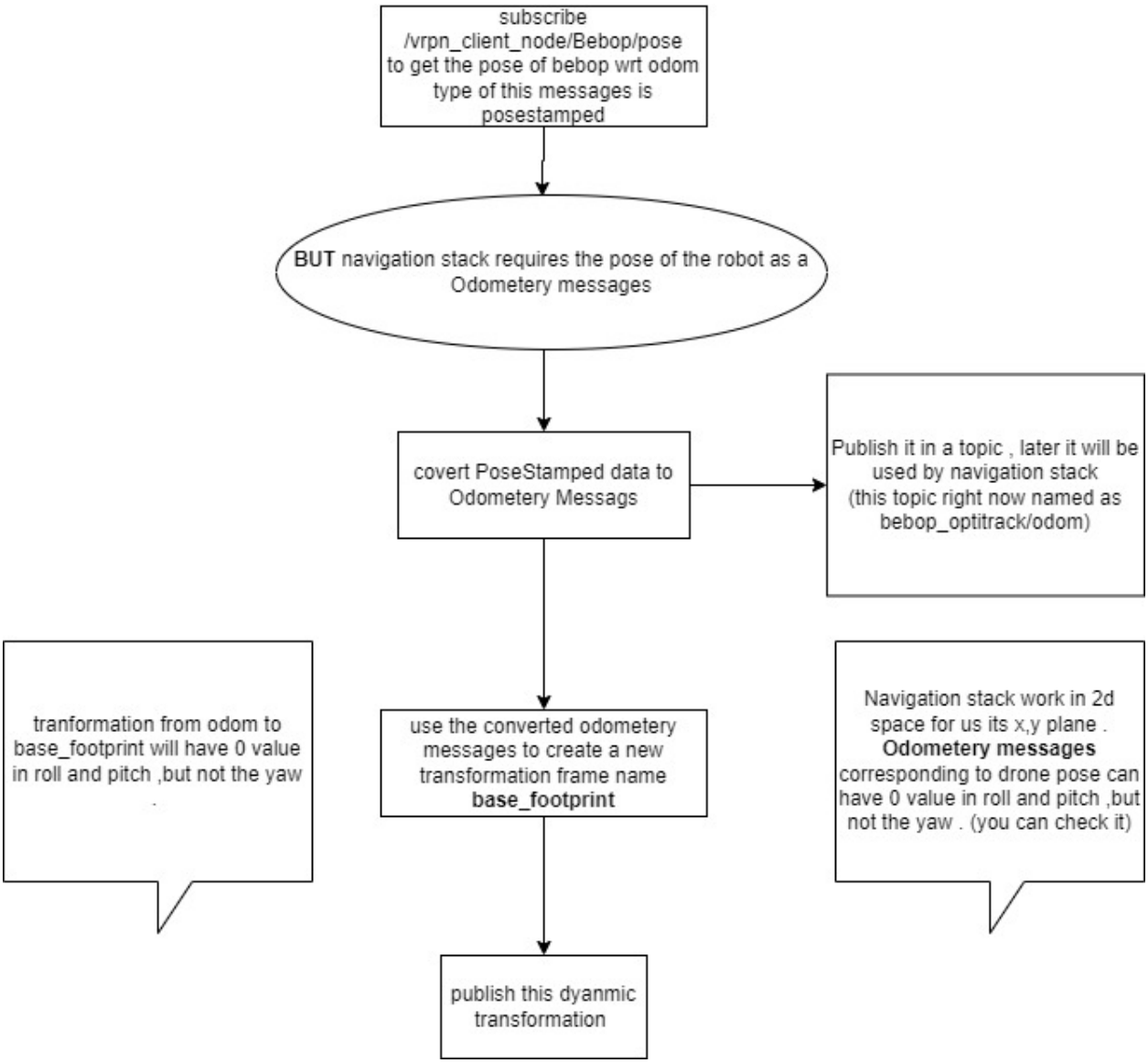
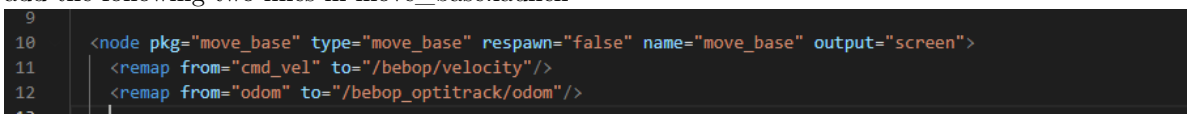


Figure 2: drone_pose_steps

add navigation stack

- from lab9 copy your bebop_navigation package in this workspace
- add the following two lines in move_base.launch
- 

FOR GRADING

- (not mandatory) Create a video using your mobile/screen capture showing the following
  - show rviz when you give location
  - show physical drone movement from start to goal position in real world
  - show rviz and terminal once drone reach the goal poistion
- Answer the questions asked in scripts
- Also answer the following
  - what is the use of remap parameter that we add in move_base.launch ?
  - why are we remapping those two parameters ?
- do not submit the package