navigation stack setup with optitrack for physical drone

Prajwal Thakur

2022-11-11

Steps

- Create a new catkin workspace (**must**) . (Lot of packages are common to lab8 and lab9 but contents are different , create conflict if you use lab8 or lab9 workspace)
- write the following line in your .bashrc
  - export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:path to your catkin_ws/devel/lib/parrot_arsdk
- install vrpn_client to connect with optitrack system
  - sudo apt-get install ros-noetic-vrpn-client-ros
- copy the provided packages from the nav_packages_real_drone.zip to your workspace
- install following packages
  - sudo apt-get install libavahi-client-dev
- build the package , **important** : building packages might require very high resource . I suggest using -j tag to run only few parallel jobs
- do : catkin build -j2
- open 4 tabs in terminal and source the work-space in each of them

Once above steps are done , do the following steps in presence of instructor

to get the poses of object from optitrack

roslaunch vrpn_client_ros sample.launch server:=192.168.1.134

to publish the velocity command to drone

rostopic pub /bebop/velocity geometry_msgs/Twist

get rqt_frame

rosrun rqt_tf_tree rqt_tf_tree

takeoff

```
rostopic pub /bebop/takeoff std_msgs/Empty “{}”
```

bebop driver conroller

```
roslaunch bebop_driver bebop_node.launch
#landing rostopic pub /bebop/land std_msgs/Empty “{}”
#tf_Tree
rosrun tf tf_echo Bebop base_footprint
```