

CSCI5551 Homework 2 and Extra Credit 2

Prerequisites

Before you begin, ensure you have the ROS1 Noetic install on you PC. You can follow the [ROS Environment Setup](#) for ROS installation.

Introduction

The purpose of this assignment is to give you experience writing ROS code. The stuff covered in this assignment will likely be involved in some way in many of your projects.

In this assignment you will write:

- ROS package
- ROS Publisher, Subscriber, Launch File
- Utilizing robot in Gazebo simulation
- *Extra Credit 2*

Terminal Configuration

Make sure to add the following lines to the end of your terminal config file located at ~/.bashrc so ROS commands work correctly. You only need to do this once, as every new terminal will run ~/.bashrc when launched.

```
# without adding to the .bashrc, you need to do the following everytime when  
you open a new terminal window.  
source /opt/ros/noetic/setup.bash  
source /home/$USER/catkin_ws/devel/setup.bash
```

Required Packages

Once you've installed ROS in Ubuntu enter the following terminal commands to navigate to your home directory and create a new catkin workspace.(you can change the workspace folder name)

```
cd ~  
mkdir -p catkin_ws/src  
cd catkin_ws/src
```

ROS packages:

Download/Clone the git repo associated with the package into the **YOU_WORKSPACE/src** folder:

- turtlebot3: <https://github.com/ROBOTIS-GIT/turtlebot3>
- turtlebot3_simulation: https://github.com/ROBOTIS-GIT/turtlebot3_simulations
- turtlebot3_msgs: https://github.com/ROBOTIS-GIT/turtlebot3_msgs

after git clone:

```
cd ../  
catkin_make
```

catkin_make will tell you if you are missing any packages. Look at the output and install the missing package(s) using the same method from above.

if the installation success, you will see:

```
...  
[ 95%] Built target flat_world_imu_node  
[ 96%] Linking CXX executable  
/home/shi00317/Downloads/ws/devel/lib/turtlebot3_bringup/turtlebot3_diagnostics  
[ 98%] Linking CXX executable  
/home/shi00317/Downloads/ws/devel/lib/turtlebot3_gazebo/turtlebot3_drive  
[ 98%] Built target turtlebot3_diagnostics  
[ 98%] Built target turtlebot3_drive  
[100%] Linking CXX executable  
/home/shi00317/Downloads/ws/devel/lib/turtlebot3_fake/turtlebot3_fake_node  
[100%] Built target turtlebot3_fake_node
```

verifying-the-installation

Open a terminal and running following command. You should see a Gazebo simulation running and there is a turtlebot in the environment.

```
export TURTLEBOT3_MODEL=burger
roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

Getting Started

Both Homework 2 and Extra Credit 2 are due in Nov 29th midnight.

Create hw_pkg folder in your workspace.

1. Turtlebot

- Using LiDAR sensor data in turtlebot to achieve obstacle avoidance ability in gazebo simulation. In the simulation environment, the turtlebot will keep moving forward if there is no obstacle in front of it. otherwise, it will automatically turn right until there is no obstacle. (30 points)
- A launch file, which could launch turtlebot_teleop and the obstacle avoidance together. Create a directory called **launch** under package directory and add .launch under **launch** directory. (20 points)

You can run two publisher by using:

```
roslaunch hw_pkg Your_Launch_File
```

2. Kinova Gen3 Robot Manipulator

- Download/Clone Kinova manipulator package and following the installation steps in the repo: https://github.com/Kinovarobotics/ros_kortex.
- Running Kinova manipulator Gazebo simulation environment

```
roslaunch kortex_gazebo spawn_kortex_robot.launch gripper:=robotiq_2f_85
```

- For Apple Silicon Macbook, The ros_kortex package is not support arm-cpu very well especially for Apple Silicon. Please install the docker engine in your Virtual Machine (refer to **ROS Environment Setup**) and pull the pre-build docker image.

```
# Download the pre-build docker image
docker pull shi00317/ros_kortex:test
# Running the image as container
xhost +local:root
```

```
docker run -it --rm -e DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix --net=host -e ROS_MASTER_URI=http://172.17.0.1:11311 -e ROS_IP=172.17.0.1 shi00317/ros_kortex:test
```

```
#inside of the image  
source ros_entrypoint.sh
```

- Create a script to use *moveit_commander* python package to collect 5 trajectories and save it. You need decide gripper(end-effector) start pose and end pose. (25 points)
- Load 5 trajectories you collected and execute those in the gazebo simulation. (25 points)

Extra Credit 2

- A ROS node subscribes to the turtlebot gazebo "scan" topic. Get the float32[] ranges and publish it to topic "ranges". User can check published data by this:

```
rostopic echo /ranges
```

Submission

- **Both Homework 2 and Extra Credit 2 are due in Nov 29th midnight.**
- Upload a zip file contain your workspace folder to the class Canvas.
- A demo video to demonstrate obstacle avoidance in turtlebot simulation environment and five trajectories execution in Kinova robot simulation environment.
- For extra credit code, if you choose to do it, to the separate Extra Credit 2 assignment on Canvas using the naming format: extra-credit-2-UMNEmail.py

Copying from others or the internet, or using ChatGPT or other AI tools is viewed as cheating. In this case, you will get an "F" in the class and pertinent University policies will be followed.