

ROB314 – Session 1 – Ex. 3

Husky and Gazebo

Theory

- ROS Architecture
- ROS Master, Nodes, and Topics
- Console Commands
- Catkin Workspace and Build System
- Launch Files

Exercise

Get familiar with ROS by inspecting the simulation of a Husky robot.

1. Preparation:

```
sudo apt update
sudo apt install ros-noetic-position-controllers ros-noetic-effort-controllers
ros-noetic-joint-state-controller
```

2. Test Gazebo 11:

```
gazebo
```

To exit the program: Ctrl + C in the terminal. It may take a while...

3. Setup the Husky simulation

```
sudo apt-get install ros-noetic-husky-simulator
```

4. Launch a simulation, for example :

```
roslaunch husky_gazebo husky_playpen.launch
```

It will take a long time to start, because the simulator has to download resources from the Gazebo servers.

You can move the visualization with the left mouse button or shift + left mouse button.

And check the created nodes and their topics with:

```
roscd husky_gazebo
rosnode list
rostopic list
rostopic echo /a_nice_topic (for example /imu/data, /navsat/fix, etc.)
rostopic hz /a_nice_topic
rqt_graph
```

For more information take a look at the slides or go to:

<http://wiki.ros.org/rostopic>

<http://wiki.ros.org/rosnode>

5. **Have a look at the file** `husky_playpen.launch`. To find this file, you can use the command `roscd husky_gazebo`.

Look at the other launch files included in `husky_playpen.launch`

6. **Command a desired velocity** to the robot from the terminal (`rostopic pub [TOPIC]`)

Hint 1: You can use this topic `/husky_velocity_controller/cmd_vel`

Hint 2: If the robot stops again after sending the velocity command, specify the rate of the publisher. Check out `rostopic pub -help`.

7. Create a **catkin workspace** `catkin_ws` as described in the lesson.
8. Use the **teleop_twist_keyboard** package to control your robot with the keyboard. Find it online (github) and compile it from source!

Use `git clone` to clone the repository to the folder `~/catkin_ws/src`.

Compile with the dedicated ROS tool...

For a quick git overview see: http://rogerdudler.github.io/git-guide/files/git_cheat_sheet.pdf

Don't forget to do a:

```
source ~/catkin_ws/devel/setup.bash
```

9. Create a new directory "launch" in `~/catkin_ws/src/teleop_twist_keyboard/`. Inside, write a launch file, called for example `husky_robotcup.launch` with the following content :

- husky simulation with another world:

Include the `empty_world.launch` file and change the `world_name` argument, e.g.

`/usr/share/gazebo-11/worlds/robotcup14_spl_field.world`

- `teleop_twist_keyboard.py` node

Kill everything and run this launch file with a **roslaunch** command. The first time, it will take a long time to start...

After that, you can control the robot with the keyboard.