2. Assignment, Introduction to Robotics WS17/18

Prof. Daniel Göhring, Martin Dreher, Jakob Krause
Institut für Informatik, Freie Universität Berlin

Submission: online until Tuesday, 07 Nov 2017, 11:55 a.m.

Upload your python file and video file, use comments in you code.
Only one member of the group must submit these 2 files.
Do not copy solutions to other groups.
Every group shall contain two people.
Only submissions via KVV will be accepted.

In your last assignments you used the following commands to
steer the car and to control its velocity - as an example:

```
rostopic pub /manual_control/speed       std_msgs/Int16 "data: -10"
rostopic pub /manual_control/steering std_msgs/Int16 "data: -8"
```

These commands were executed in your shell. Let’s use ros-nodes!

1. First test drive (10 Points)

This time you should write a python rosnode which publishes these two topics. As a result your car should move straight and turn, so that finally your car moves on a square with arcs as corners and stops. So your car shall move straight and turn, etc., altogether 4 straights and four turns. The square does not have to be perfect but finally the car should stop in the vicinity of where it started.

As submission you will have to upload:

1. a video (no larger than 5 MB, no longer than 30s) which shows how the car moves in the simulator and performs all the required motions to move on a square-like shape.
2. The python file “talker.py”. Comment the code in that file to explain what your code is supposed to do.

The video **must**:

- show (e.g. on one console) your group name - in the video
- not be edited, you need to show everything in one take (no cutting)
- use some screen-capturing software, e.g., recordmydesktop. If for some reason, this does not work for you, use your smartphone or another camera.
- not be larger than 5 MB and no longer than 30s. That said, the car has to perform its motions in at most 30 s.
- feel free to use video compression tools, e.g. avconv.
- be in .ogv or .mp4 format

Hints:

You shall understand how nodes, topics and publishers work.therefore, read the following tutorial:

http://wiki.ros.org/ROS/Tutorials/CreatingPackage

Now go to the simulated model car folder:

```
cd ~/seat/model_car_3/catkin_ws/src
```

Create your first own package, named task02_simple motion:

```
catkin create pkg task02_simple_motion std_msgs rospy
```

go into the folder task02_simple_motion and start:

```
catkin build
```

Now you need to source again your setup bash:

```
source ~/seat/model_car_3/catkin_ws/devel/setup.bash
```

now you can roscd into your new package:

```
roscd task02_simple_motion
```

For your python file, create a folder and go into that folder:

```
mkdir scripts
cd scripts
```

Read the tutorial, how a simple python node which publishes a string works:

http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber%28python%29

Load the talker.py - file (the whole command in one line):

```
wget
https://raw.github.com/ros/ros_tutorials/kinetic-devel/rospy_tutorials/001_talker_listener/talke
r.py
```

Make the talker.py executable with

```
chmod +x talker.py
```

start the node:

rosrun task02_simple_motion talker.py

You will see printouts on your console, also you can see, that a /chatter topic is being published, when you type

rostopic list / chatter

Now you need to modify the publisher (variable pub) in order to send steering angle topics - so you need to change the /chatter topic into the topic for steering (see above). This topic also uses a Int16 instead of a String, modify accordingly.

Create a second publisher to publish velocity commands.

Now use both commands to move your car straight for a couple of seconds, turns right (or left), moves straight and so on.
At the end the car has to stop.

As a general hint, if you open multiple terminals, make sure that you source the right files (setup.bash) in each console window.

You do not need to use any localization data or odometry to solve this task.