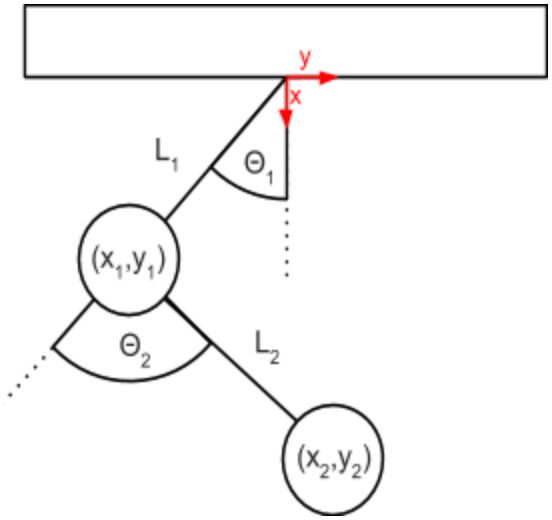
2. Assignment, Introduction to Robotics WS16/17

Prof. Daniel Göhring, Zahra Boroujeni
Institut für Informatik, Freie Universität Berlin
Submission: online until Tuesday, 15 Nov 2016, 11:59 a.m.

Please summarize your results (images and descriptions) in a pdf-document and name it "RO-02-<surnames of the students - group name>.pdf". There should not be any source code in the pdf document.
By the end of this class you will need 50% of points to be allowed to write the final exam.

1. Coordinate Transformation (2 Points):



a) (1 Point) Given the above double pendulum. Give the 4 equations to calculate x_1,y_1, x_2 und y_2 with respect to Θ_1 and Θ_2 . L_1 and L_2 are constant.
b) (1 Points) Write the partial derivatives (4 equations) for position coordinates x2 and y2 from a) with respect to Θ_1 and Θ_2, e.g., partial derivative of x2 with respect to Θ_1 , etc.

2. Rotation Matrix as an Operator (2 Points):

a) (0.5 Points) How many random numbers are necessary to generate a non-trivial random rotation matrix R? Which elements need to be defined? Please provide an example.
b) (0.5 Points) Use the translation vector of (3, 2, 5) to generate a homogeneous transformation matrix in combination with your matrix from a) in a non-trivial way.
c) (1 Point) Now use this matrix as an operator to manipulate a point (1,2,3), therefore provide the calculation equation briefly.

3. Working with the Car (6 Points):

Use the **new(!)** online doodle-poll to reserve a time slot for you
http://doodle.com/poll/t7k87df7iaan3tyh

A- Change speed of the motor from 0 rpm to 500 rpm. Draw the graph of the motor speed vs time. And measure delay of the motor to reach the desired speed (500 rpm)

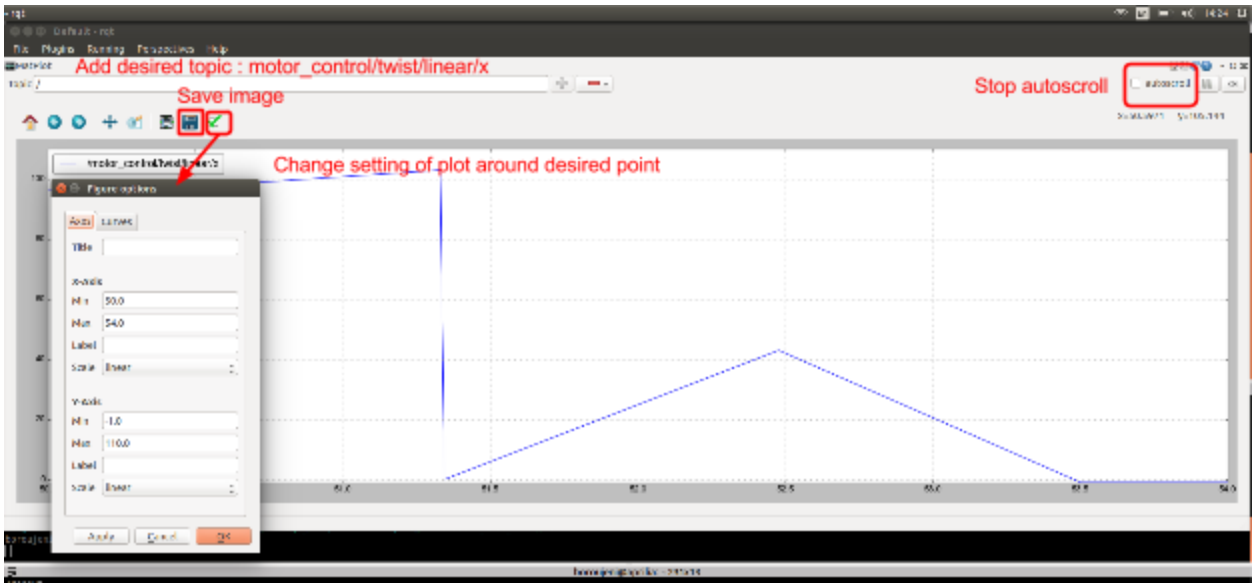
Steps you may need:

- Turn on the car , push the green button twice to see the green light https://github.com/AutoModelCar/AutoModelCarWiki/wiki (Powering on and off section)
- Read the http://wiki.ros.org/rostopic (1.8 rostopic pub)
- You can publish 500 rpm as a desired speed once using terminal command:

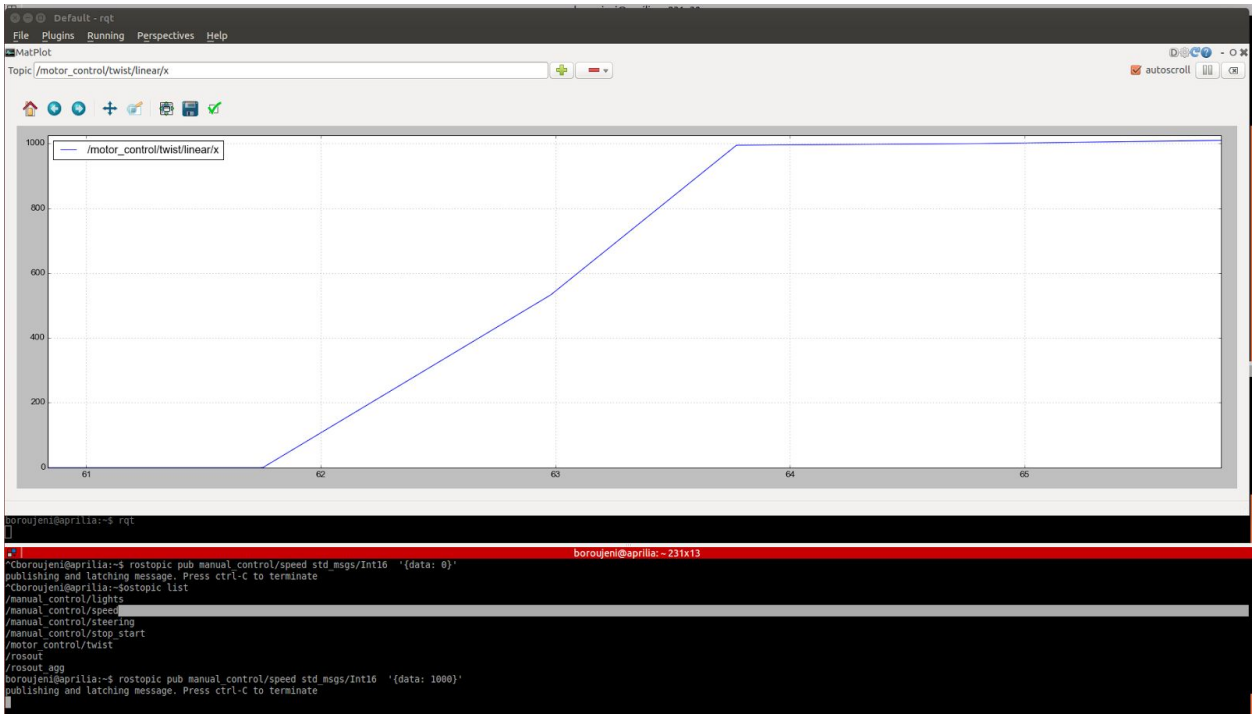
  rostopic pub manual_control/speed std_msgs/Int16 '{data: 500}'

- Read the rqt_graph wiki : http://wiki.ros.org/rqt_plot
- Run rqt_graph or (rqt -> plugins -> visualization -> plot)
- Plot /motor_control/twist/linear/x topic vs time
- Change setting of rqt_gragh to measure the delay of the motor to reach the desired speed

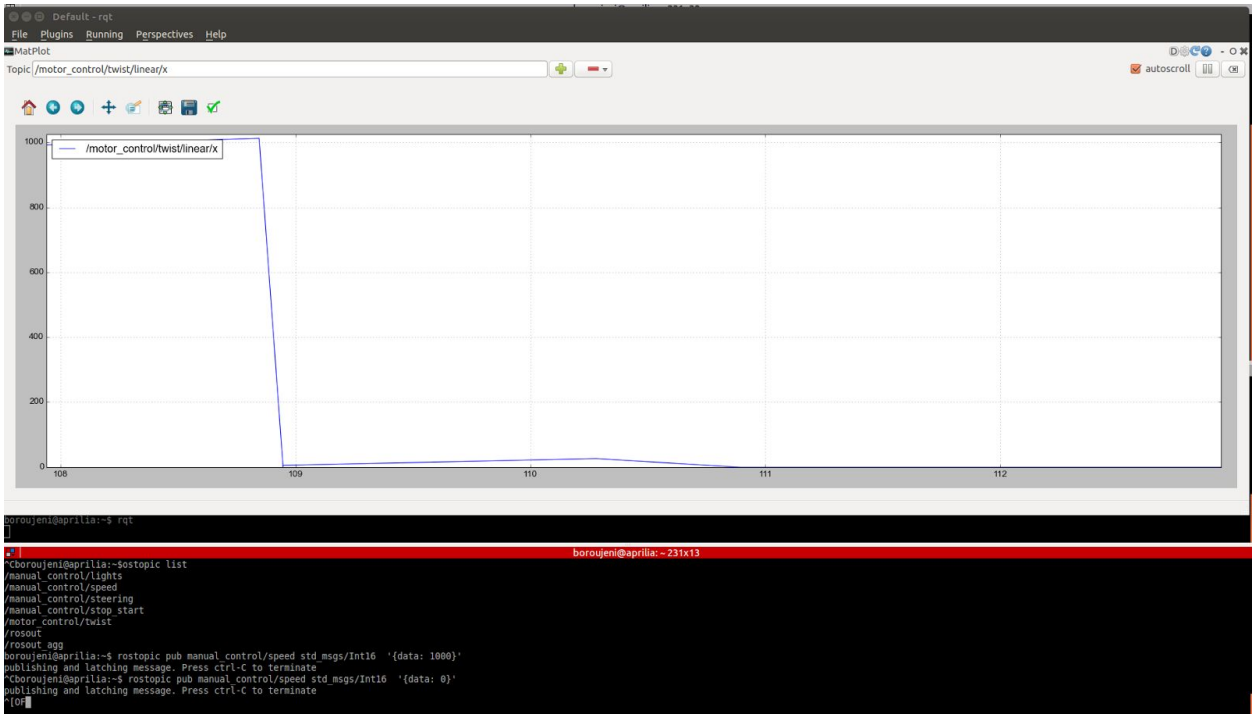
Examples:



Pic1: how to use rqt/plugins/visualization/plot/ : change the speed from 100 rpm to 0 rpm



Pic2: change speed of motor from 0 rpm to 1000 rpm (delay around 2 s)



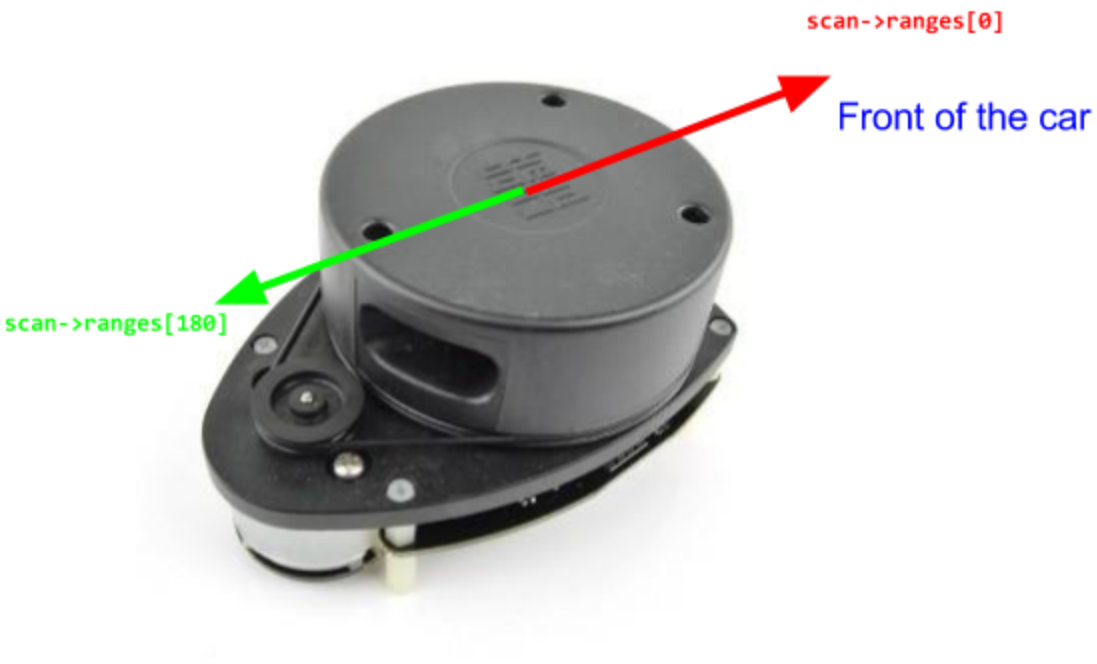
Pic3: change speed of motor from 1000 rpm to 0 rpm (delay around 100 ms)

B) Use a box as an obstacle. Move the box toward the car (from 2 meter away). Subscribe to the laser scanner (read the ranges in front of the car) and change the motor speed based on

obstacle distance (decrease the speed while distance is decreasing) . Draw the motor speed and obstacle distance vs time (both in a same plot which we can compare). Or if you can plot motor speed vs obstacle distance.

Steps you may need:

- Read the ros wiki for publishing and subscribing to the topics using c++: http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber(c%2B%2B)
- Subscribe to /scan topic :



- Publish desired speed on manual_control/speed based on the /scan->ranges
- Publish distance between the obstacles and the car (you can use average of ranges between 140 and 220 as an obstacle distance) /obstacle_distance
- Use rqt_graph to plot /obstacle_distance (or just one beam /scan/ranges[0]) and /motor_control/twist/linear/x

Hint: look at code below would help you to implement the lidar assignment: https://github.com/AutoModelCar/model_car/blob/master/catkin_ws/src/auto_stop/src/auto_stop.cpp