

POLITECNICO DI MILANO


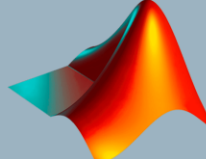
AUTONOMOUS VEHICLES

S. Arrigoni



POLITECNICO
MILANO 1863

Software requirements

 ROS +  MATLAB®

software architecture

- Initialize gazebo with turtlebot3_empty_world.launch

```
$ export TURTLEBOT3_MODEL=burger  
$ roslaunch turtlebot3_gazebo turtlebot3_empty_world.launch
```

- Initialize MATLAB/Simulink and connect it to ROS.

roslaunch

roslaunch('ip_address of the master')

(if using Virtual Machine)

Alternative software solution

To install and start the turtlesim:

```
$ sudo apt-get install ros-$(rosversion -d)-turtlesim
```

Run turtlesim:

```
$ rosrun turtlesim turtlesim_node
```

- It is **already** part of **ROS (full desktop)**, in case you don't have it follow the steps reported and install it!



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



Assignment II

What we expect from you

Starting from a list of waypoints provided (waypoints.txt):

define a **feedback control** in order to move the #robot from initial position through all waypoints and to stop it at the last one.

List of suggested steps

- Create a simulink block:
 - Incorporate waypoints list
 - Subscribe to #robot position (/odom or /pose)
 - Decide the waypoint to reach
 - Compute control command to publish
 - Stop if last point is reached

Feedback control

- How to reach a waypoint? (easy way)
 - Consider your distance to the waypoint
 - Consider the direction between you and the waypoint
 - When close to the waypoint, fix your heading
- Control constraints
 - $\max |v| = 0.2;$
 - $\max \text{dist to waypoints} = 0.05$
 - $\max |\text{yaw rate}| = 0.4;$

HINT: just proportional terms

Results

- What is mandatory for the report?
 - Plot of #robot trajectory and waypoints
 - Plot of velocity time evolution
 - (save a .bag with #robot position, control command)



Bonus Request

extra A

Waypoint list provided by an external publisher
(i.e python_node, command line, ...):

- Create a publisher
- Use message type: "***geometry_msgs/PoseWithCovariance***"
- Insert waypoints into ***covariance*** field (array[36] of float64)
- Modify your simulink in order to subscribe to it

extra B

Try to implement a **more complex** feedback control logic
(take inspiration by lectures or literature)

That's it for today...

See you next time!

S. Arrigoni



POLITECNICO
MILANO 1863