Last update: 19-07-2019

240AR060 - Introduction to Ros

Coordinating unit: 240 - ETSEIB - Barcelona School of Industrial Engineering

Teaching unit: 707 - ESAII - Department of Automatic Control

Academic year: 2019

Degree: MASTER'S DEGREE IN AUTOMATIC CONTROL AND ROBOTICS (Syllabus 2012). (Teaching unit Optional)

ECTS credits: 4,5 Teaching languages: English

Teaching staff

Coordinator: Rosell Gratacos, Joan

Others: Rosell Gratacos, Joan
Rosales Gallegos, Carlos J.

Opening hours

Timetable: To be defined

Prior skills

The student should have basic skills in C++ programming as well as linux common tools and commands. An overall understanding of software processes involved in robotics will be welcomed.

Teaching methodology

The teaching methodology will combine lectures together with supervised exercises based on the current ROS version and tools. All classes will be organized with the theoretical sessions at the beginning and practical exercises and team work at the end. The initial part will consist on the explanation of theoretical concepts by the lecturer, promoting the active participation of students. The practical part will be focused on the student's solving skills. The main theoretical concepts will be shown in practical simulation examples and finally on a real robot test.

Learning objectives of the subject

The objective of this course is to introduce students in the use of ROS as a powerful robotics tool. Specifically a familiarization with the middleware concept and the software structure of a robot. There will be a special emphasis on sensing and control of robots using ROS, both in simulation and in real environments.

Learning Outcomes:
Learn how to setup a Linux O.S. environment to work with ROS.
Understand the ROS communications architecture.
Use ROS in the different process layers, from sensing to control or actuation.
Implement simple ROS projects with both simulation and real robots.

Mandatory contents:
Install and setup ROS in a native O.S. Linux (Ubuntu).
Know and understand the internal procedures of ROS and its modules functionalities (master,nodes, and so on).
Identify and use the ROS tools and formats related to the internal communication between nodes (topics, actions, services,...).
Use ROS visualization and debugging tools.
Design and program C++ algorithms using ROS as a middleware.

Last update: 19-07-2019

240AR060 - Introduction to Ros

Use debugging tools to verify the compilation and the algorithm functionalities.
Configure and use a simulation environment with the designed algorithms.
Managing acquisition, analysis and display of data obtained from different sensors using ROS.
Manage and send control commands to a robot using ROS, both using simulation and real settings.

Study load

Total learning time: 112h 30m	Hours large group:	27h	24.00%
	Hours small group:	13h 30m	12.00%
	Self study:	72h	64.00%

Last update: 19-07-2019

240AR060 - Introduction to Ros

Content

1. ROS Basic concepts

Learning time: 5h
Practical classes: 3h
Self study : 2h

Description:
1.1. Introduction
1.2. ROS core components
1.3. Applications
1.4. Install instructions
1.5. ROS command-line tools

2. Development Tools

Learning time: 5h
Practical classes: 3h
Self study : 2h

Description:
2.1. Programming
2.2. Building executables
2.3. The ROS build system
2.4. Good practices
2.5. Version control using GIT

3. Communications using topics

Learning time: 5h
Practical classes: 3h
Self study : 2h

Description:
3.1. An example: The package agitr_chapter3
3.2. A publisher program
3.3. A subscriber program
3.4. Standard and common messages

Last update: 19-07-2019

240AR060 - Introduction to Ros

4. The launch utility	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
4.1. Using launch files
4.2. Understanding launch files
4.3. Graph resource names
4.4. Managing names in launch files
4.5. ROS parameters

5. Communications using services	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
5.1. Services
5.2. The package agitr_chapter8
5.3. A client program
5.4. A server program
5.5. Standard services
5.6. Defining non-standard services

6. Tools	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
6.1. The tf tool
6.2. Robot Modeling and visualization tools
6.3. The rosbag Tool
6.4. The rqt tool

Last update: 19-07-2019

240AR060 - Introduction to Ros

7. Communications using actions	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
7.1. Working with ROS actionlib
7.2. Building and running a simple example
7.3. The ROS action server
7.4. The ROS action client
7.5. The pan-tilt example

8. Simulation - basic issues	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
8.1. Gazebo basics
8.2. Integration to ROS
8.3. Configuring launch files
8.4. ROS-aware Gazebo plugins
8.5. Tunning URDF models

9. Simulation - sensors	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
9.1. Available ROS plugins
9.2. The camera ROS plugin
9.3. The depth camera ROS plugin
9.4. ROS plugins for some other sensors

Last update: 19-07-2019

240AR060 - Introduction to Ros

10. Robot control	Learning time: 5h Practical classes: 3h Self study : 2h

Description:
10.1. ros_control overview
10.2. Controllers
10.3. Hardware Abstraction Layer
10.4. Using ros_control in Gazebo

Case study	Learning time: 15h Practical classes: 9h Self study : 6h

Description:
Definition of the solution
Sensing module
Planning module
Action module

Qualification system

The acquired competences and capabilities will be assessed on the basis of three qualification grades: exercises (20%), deliverable (20%) and final project (60%).
Re-evaluation: new final project (60%).

Last update: 19-07-2019

240AR060 - Introduction to Ros

Bibliography

Basic:

Jason M. O'Kane. A gentle Introduction to ROS [on line]. S.L.: CreateSpace Independent Publishing Platform, 2013 [Consultation: 27/06/2019]. Available on: <https://cse.sc.edu/~jokane/agitr/agitr-letter.pdf>. ISBN 9781492143239.

Others resources:

ROS wiki page: http://wiki.ros.org/
ROS tutorials: http://wiki.ros.org/ROS/Tutorials/
Gazebo tutorials: http://gazebosim.org/tutorials/
Catkin tutorials: http://jbohren.com/tutorials/
Git tutorial: https://try.github.io/, https://guides.github.com/activities/hello-world/
ROS cheatsheet: https://github.com/ros/cheatsheet/releases/download/0.0.1/ROScheatsheet_catkin.pdf /
Catkin tools cheatsheet: https://catkin-tools.readthedocs.io/en/latest/cheat_sheet /
Git cheatsheet: http://rogerdudler.github.io/git-guide/files/git_cheat_sheet.pdf

Hyperlink

Introduction to ROS: online tutorials

https://sir.upc.edu/projects/rostutorials/index