Overview

*Marine Robotics and ROS* is a graduate level course in applied mobile robotics with a strong emphasis on control of marine robots using the Robot Operating System (ROS).

**Marine Robotics:** The course will focus on applying specific Guidance, Navigation, and Control (GNC) solutions for a known platform. I.e. the robot model is assumed known and the student is expected to develop autonomous and/or semi-autonomous behaviors.

**ROS:** At its core, ROS is a publish-and-subscribe network that provides a “flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.”¹ Most ROS tools libraries and add-ons are written in python, or C++, but ROS does provide APIs for other programming languages (e.g. Matlab).

This is an applied, project-based course. **You are expected to learn heuristically and to take control, as grad students should, of your learning objectives.**

Objectives

What do I want the students to be able to accomplish at the end of this course?

- **GNC**
  - Identify/Develop/Apply a global control architecture/block diagram.
  - Develop, simulate, and apply a solution for all three aspects of GNC.
    - Utilize an appropriate model to develop.
    - Tune and modify for real world application.

- **ROS**
  - Can develop a publisher, subscriber, service, action and knows when to utilize each.

- **Publication**
  - Can communicate effectively in an academic context.

¹ [http://www.ros.org/#](http://www.ros.org/#)
**Required Resources:**

- ROS Capable computer
  - Distribution:
    - ROS 2 Humble Hawksbill

**Assignments and Grading**

Late work will not be accepted.

<table>
<thead>
<tr>
<th>Participation</th>
<th>15%</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Class discussions</td>
<td></td>
</tr>
<tr>
<td>- Quizzes</td>
<td></td>
</tr>
<tr>
<td>- Assignments</td>
<td></td>
</tr>
<tr>
<td>- Tutorials</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project 1</th>
<th>20%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project 2</td>
<td>25%</td>
</tr>
<tr>
<td>Project 3</td>
<td>40%</td>
</tr>
</tbody>
</table>

**References:**

- **Robotics**
    - [http://hades.mech.northwestern.edu/images/7/7f/MR.pdf](http://hades.mech.northwestern.edu/images/7/7f/MR.pdf)
    - [https://www.youtube.com/playlist?list=PLggLP4f-rq02vX0QQQ5vrCxbJrzamYDfx](https://www.youtube.com/playlist?list=PLggLP4f-rq02vX0QQQ5vrCxbJrzamYDfx)
    - [https://www.youtube.com/playlist?list=PLMrJAkhIeNNR20Mz-Vpzgfs5zrYi085m](https://www.youtube.com/playlist?list=PLMrJAkhIeNNR20Mz-Vpzgfs5zrYi085m)
    - [https://www.youtube.com/channel/UC-45kyxsA0XwgDTulgpa9kw/videos](https://www.youtube.com/channel/UC-45kyxsA0XwgDTulgpa9kw/videos)

- **ROS**
  - General
    - [https://www.openrobotics.org/](https://www.openrobotics.org/)
    - [www.ros.org](http://www.ros.org)
    - [https://navigation.ros.org/](https://navigation.ros.org/)
    - [https://moveit.ros.org/](https://moveit.ros.org/)
• Conventions
  o http://www.ros.org/reps/rep-0000.html
  o http://wiki.ros.org/ROS/Patterns/Conventions

• Style guides
  o http://wiki.ros.org/StyleGuide
  o http://wiki.ros.org/DevelopersGuide
  o http://wiki.ros.org/CppStyleGuide
  o http://wiki.ros.org/PyStyleGuide
  o http://www.ros.org/reps/rep-0008.html

• Best Practices
  o Never edit files in /opt/ros/…
  o http://wiki.ros.org/ROS/Patterns
  o https://github.com/leggedrobotics/ros_best_practices/wiki