AMR 2015/2016: Final Projects

A final project consists of:

- reading some papers, notes or documentation on a specific subject
- performing simulations or numerical tests on a software platform (MATLAB or V-REP)
- writing a report
- giving a presentation (with slides)

As a rule, each project must be carried out by a group of **3 students**. Projects are assigned to groups on a FIFO basis. Send me an e-mail message (directly, not through the Google Group) specifying a list of **3** projects in which your group is interested, with an order of priority, and the composition of the group. 1 or 2-student groups can also apply, but I reserve the right to merge them into a larger group. The deadline for applying for a project is **May 27**. Late applications will not be accepted.

Note that four larger projects are **shared** with **Robotics 2**; this means that they are simultaneously valid as final projects for both courses. Two of these shared projects are in the list below (with numbers 1 and 2); two more are in the project list provided by prof. De Luca. If your group is applying for a list of shared projects, please send an e-mail message to **both** me and De Luca.

Once your group has been assigned a project, we will set up a meeting to discuss the project in detail. There will be three deadlines for submitting your projects: **June 30**, **September 30**, **December 31 (2016)**. To submit your project, send me an e-mail with the report. Once a deadline is passed, I will fix a common date for presenting all the projects completed during the associated time window.

This is the list of the available projects:

1. Kinematic control alternatives for humanoid gait execution **(shared with Rob 2)**

2. Motion planning between equilibria for a BallBot **(shared with Rob 2)**

3. Formation control of a team of mobile robots

4. Predictive task control for a nonholonomic mobile manipulator

5. RRT with Poisson sampling

6. Bug algorithms for a unicycle

In the following pages a short description of each project is given.

1. Kinematic control alternatives for humanoid gait execution (shared with Rob 2)

Synopsis
The aim of this project is to test some kinematic control alternatives for executing a walking gait in a humanoid robot. In particular, in the situation of interest a stable CoM trajectory is computed from a ZMP trajectory (footsteps). A composite task made of the CoM trajectory and the swing foot trajectory is then sent to the kinematic control, which must compute joint velocities that realize the task asymptotically. The objective of the project is (1) to assess the relevance of the transient tracking error (2) to compare two possible methods for reducing such error, i.e., (2a) perform a pre-actuation in order to assume the initial conditions that guarantee exact tracking, and (2b) use second-order kinematic control to introduce an explicit dependence of the joint commands on the task velocity error. Depending on the expertise of the group, the implementation platform may be MATLAB or V-REP.

Reading material
Notes on stable gait generation and on kinematic control for gait execution
MATLAB or V-REP software documentation

2. Motion planning between equilibria for a BallBot (shared with Rob 2)

Synopsis
The BallBot is an underactuated robot consisting of a tall body balancing on a spherical wheel. Planning motions for underactuated robots is challenging due to the fact that the number of control inputs is smaller than the number of degrees of freedom. The objective of this project is (1) to derive a dynamic simulator for the BallBot (2) to implement a task-constrained motion planner for this system. In particular, the latter needs to work at the third-order (snap) level to guarantee that the final configuration reached by the robot is an equilibrium. The implementation platform is V-REP.

Reading material
Papers on the BallBot
Cefalo, Oriolo, “Task-Constrained Motion Planning for Underactuated Robots,” ICRA 2015
Notes on third-order planner
V-REP software documentation

3. Formation control of a team of mobile robots

Synopsis
The goal of this project is to simulate a group of mobile robots having all unicycle or car-like kinematics, and to design schemes for controlling tasks related to their formation (barycenter, variance, etc). Two possible approaches will be considered for task control. The first is based on input-output feedback linearization with the addition of null-space commands. The second is based on dynamic feedback linearization. A critical comparison based on simulations should be performed. Depending on the expertise of the group, the implementation platform may be MATLAB or V-REP.

Reading material
De Luca, Oriolo, Robuffo Giordano, “Kinematic Control of Nonholonomic Mobile Manipulators in the Presence of Steering Wheels,” ICRA 2010
Notes on formation control
MATLAB or V-REP software documentation

4. Predictive task control for a nonholonomic mobile manipulator

Synopsis
The goal of this project is to implement and test a task controller for redundant robots based on the model predictive control approach. In particular, the application platform will be a nonholonomic mobile manipulator composed by a unicycle vehicle equipped with an elbow-type 3R arm. The idea is to use the predictive control paradigm to compute in real-time joint velocities that are optimal over a certain time horizon (rather than at a single instant). The generated motions are expected to exploit more effectively the kinematic redundancy of the mechanism. Depending on the expertise of the group, the implementation platform may be MATLAB or V-REP.

Reading material
Papers on model predictive control
MATLAB or V-REP software documentation

5. RRT with Poisson sampling

Synopsis
This project aims at implementing and testing an RRT-based motion planning algorithm that uses the maximal Poisson disk sampling scheme. The planner exploits the free-disk property of the maximal Poisson-disk samples to generate nodes and perform tree expansion and uses an adaptive scheme to generate more samples in challenging regions of the configuration space. A performance assessment of the method should be made with respect to the classical RRT over benchmark examples. Depending on the expertise of the group, the implementation platform may be MATLAB or V-REP.

Reading material
Park, Pan and Manocha, "Poisson RRT", ICRA 2014
MATLAB or V-REP software documentation

6. Bug algorithms for a unicycle

Synopsis
Most motion planning methods assume the complete knowledge of the environment. In the absence of a map, the robot must discover the world through its sensors and navigate on the basis of this incremental information. Bug algorithms are a class of planning methods that apply to robots with very basic sensor capabilities, like detecting contact with an obstacle and moving along its contour. Interestingly, they are complete, in the sense that they find a solution whenever one exists. The objective of this project is to realize an implementation of this approach on a unicycle robot. Depending on the expertise of the group, the implementation platform may be MATLAB or V-REP.

Reading material
Paper on bug algorithms
MATLAB or V-REP software documentation