FIELD AND SERVICE ROBOTICS
Università degli Studi di Napoli Federico II
Dipartimento di Ingegneria Elettrica e Tecnologie dell'Informazione

Instructor: Fabio Ruggiero, Ph.D. [fabio.ruggiero@unina.it]

[Updated 29/10/2020]

INSTRUCTIONS FOR THE TECHNICAL PROJECT

GENERAL RULES

- The assigned topic must be solved individually.
- A **detailed English report** must come along the produced **code** and a **video** showing the achieved results .
    - All files must be sent to the instructor's email **at least two days** before the agreed examination day.
    - The report should include all the mathematical formulations, the choices, and the plots that the student believes are of interest to the resolution of the project.
    - The report will undergo a plagiarism scan check.
- The projects are classified into two categories: *i)* wheeled robots; *ii)* aerial robots.
- The technical projects are assigned by the instructor based upon the students' surnames.
    - It is possible to ask the instructor for a change of the project **category** (not to choose a specific topic!) by and not later than the end of the course. The **request must be motivated** (*e.g.*, personal preference towards the wheeled robots instead of the quadrotors, or vice versa; preference to get an argument close to a thesis topic; ….). The instructor will then assign another project under the general criterion of balancing out all the arguments. Not all the requests will be accepted.
- There are no restrictions on the programming languages and simulation environments. Therefore, a student may use Matlab, Simulink, C++, Python, physics engines, and so on. The ROS middleware can be used as well. There is instead a restriction on the employed libraries/commands: basic mathematic libraries can be used, but it is forbidden to use libraries/commands automatically solving the problems indicated within the description of the assigned technical project. As an example, if the project requires the development of an RRT method for planning purposes, the student must implement the whole algorithm without using a library/command implementing a method to solve it.
- The points for the technical project span from 0 to a maximum of 26.
    - The instructor might assign a bonus up to 4 points to those students who present the simulation carried out in a physics engine simulator (Gazebo, CoppeliaSim,…).
    - In general, no limitations exist for the final grade regardless of the choice of presenting the results in a physics engine simulator or not.
- The instructor is always at disposition to clarify each doubt.

TECHNICAL PROJECTS ASSIGNMENT

- Surnames A-C: Project C1-P1
- Surnames D-H: Project C1-P2
- Surnames I-M: Project C2-P1
- Surnames N-Z: Project C2-P2

NOTICE FOR C2 PROJECTS

For those who want to use the Gazebo simulator, the *Characteristics* requirement of the project description below is optional. In particular, they should clone the following project and start from it: https://github.com/ruxfabio16/Drone_FSR.git

Please, before starting, read the project's readme file for prerequisites, instructions, data, and tips. The project contains a free scene with a Hummingbird quadrotor and an embedded plugin that avoids ENU and NED frames' transformations. You can edit the scene as you prefer to meet your requirements. You can put the code to implement your controller in the *quad_control* folder. The project uses the *Rotors* library, which is already included within the project to avoid having different versions of it (not the tidiest choice from a programming viewpoint, but effective).

TECHNICAL PROJECTS DESCRIPTION

1. **CATEGORY 1. PROJECT 1. (C1-P1)**
   Robot: Differential-drive robot
   Characteristics: Perform a state of the art research, or commercial research, to find out a differential drive robot from whose datasheet it is possible to extract the necessary kinematic and dynamic parameters. If some physics engine is employed, it is possible to use already existing imported models.
   Planning: A probabilistic planning method (one among PRM, RRT, or bidirectional RRT) plus the A* algorithm to find out the path. A suitable timing law must be implemented for each tract of the path, taking into account the maximum velocities of the chosen robot.
   Control: Free to be chosen among the tracking and/or the posture controllers illustrated during the course. The controller sample time is 10ms.
   Feedback: It is supposed that the exact position and orientation of the robot is at disposition.
   Scenario: The working space is a closed square environment. Three obstacles are freely placed in the environment: two of them are circular, one is a rectangular-shaped object. The dimensions of the environment and the obstacles are left free. The robot starts from an absolute configuration (position + orientation) A and must arrive in a configuration B. The obstacles and the target configurations must be placed such as the resulting path is non-trivial.

2. **CATEGORY 1. PROJECT 2. (C1-P2)**
   Robot: Differential-drive robot
   Characteristics: Perform a state of the art research, or commercial research, to find out a differential drive robot from whose datasheet it is possible to extract the necessary kinematic and dynamic parameters. If some physics engine is employed, it is possible to use already existing imported models.
   Planning: Planning via the artificial potential method, where the force field is seen as a velocity vector. The method to avoid the possible local minima is left free (artificial solutions, best-first algorithm, navigations functions, …).
   Control: Free to be chosen among the tracking and/or the posture controllers illustrated during the course. The controller sample time is 10ms.
   Feedback: Implement a passive odometric localization method to reconstruct the state of the robot from the displacement of each wheel.
   Scenario: The working space is a closed square environment. Three obstacles are freely placed in the environment: two of them are circular, one is a rectangular-shaped object. The dimensions of the environment and the obstacles are left free. The robot starts from an

absolute configuration (position + orientation) A and must arrive in a configuration B. The obstacles and the target configurations must be placed such as the resulting path is non-trivial.

3. **CATEGORY 2. PROJECT 1. (C2-P1)**

   Robot: Quadrotor

   Characteristics (optional, see the notice above): Perform a state of the art research, or commercial research, to find out a quadrotor from whose datasheet it is possible to extract the necessary kinematic and dynamic parameters. If some physics engine is employed, it is possible to use already existing imported models.

   Planning: Planning via the artificial potential method, where the force field is seen as a velocity vector. The method to avoid the possible local minima is left free (artificial solutions, best-first algorithm, navigations functions, …).

   Control: Passivity-based hierarchical controller. The controller sample time is 1ms. The controller must also deal with uncertainty about the knowledge of the dynamic parameter of the robots. In detail, the mass in the controller is underestimated by 10%, as well as the inertia parameters. Besides, an external force of 1N is continuously acting along the *x*-direction of the world inertia frame.

   Feedback: It is supposed that the exact position and orientation of the robot is at disposition.

   Scenario: The working space is a closed square environment. The maximum height is 5 meters. Three circular pillars are freely placed in the environment, and they are 5 meters tall. The radius of the pillars base and the other dimensions of the workspace are left free. The robot starts from the ground and must take off to reach an absolute configuration (position + yaw angle) A. Then, through the planning technique mentioned above, it must arrive in a configuration B. Configurations A and B must have different altitudes. Finally, the robot must land. The obstacles and the target configurations must be placed such as the resulting path from A to B is non-trivial.

4. **CATEGORY 2. PROJECT 2. (C2-P2)**

   Robot: Quadrotor

   Characteristics (optional, see the notice above): Perform a state of the art research, or commercial research, to find out a quadrotor from whose datasheet it is possible to extract the necessary kinematic and dynamic parameters. If some physics engine is employed, it is possible to use already existing imported models.

   Planning: A probabilistic planning method (one among PRM, RRT, or bidirectional RRT) plus the A* algorithm to find out the path. A suitable timing law must be implemented for each tract of the path.

   Control: Free to be chosen among controllers illustrated during the course. The controller sample time is 1ms.

   Feedback: It is supposed that the exact position and orientation of the robot is at disposition.

   Choose **one** among the following 2 scenarios

   Scenario 1: The working space is a closed square environment. The maximum height is 5 meters. Three circular pillars are freely placed in the environment: two of them are 5 meters tall, while the last one is 4.5 meters tall. The radius of the pillars base and the other dimensions of the workspace are left free. The robot starts from the ground and must take off to reach an absolute configuration (position + yaw angle) A. Then, through the planning technique mentioned above, it must arrive in a configuration B passing through a pre-determined configuration C. Such a configuration C must be located at the top of the pillar whose height is 4.5 meters. The robot must not stop in C. All the configurations A, B, and C must have different altitudes. Finally, the robot must land. The obstacles and the target configurations must be placed such as the resulting path is non-trivial.

Scenario 2: The working space is a closed square environment. The maximum height is 5 meters. Three circular pillars are freely placed in the environment, and they are 5 meters tall. The radius of the pillars base and the other dimensions of the workspace are left free. The robot starts from the ground and must take off to reach an absolute configuration (position + yaw angle) A. Then, through the planning technique mentioned above, it must arrive in a configuration B passing through a pre-determined configuration C. In this configuration C, the robot must perform a flip[1] (*i.e.*, a complete 360 degrees rotation around the *x*-axis or the *y*-axis of the body frame). Configurations A and B must have different altitudes. Finally, the robot must land. The obstacles and the target configurations must be placed such as the resulting path is non-trivial.

[1]For the flipping task you may refer to the following scientific papers, or similar ones.
[1] Y. Chen, N.O. Pérez-Arancibia, "Lyapunov-based controller synthesis and stability analysis for the execution of high-speed multi-flip quadrotor maneuvers", 2017 American Control Conference, 2017. [pdf]
[2] Y. Chen, N.O. Pérez-Arancibia, "Generation and real-time implementation of high-speed controlled maneuvers using an autonomous 19-gram quadrotor", 2016 IEEE International Conference on Robotics and Automation, 2016. [pdf]
[3] M. Cutler, J.P. How, "Actuator constrained trajectory generation and control for variable-pitch quadrotors", in AIAA Guidance, Navigation, and Control Conference, 2012. [pdf]