FIELD AND SERVICE ROBOTICS (FSR)
University of Naples Federico II
Department of Electrical Engineering and Information Technology

Instructor: Fabio Ruggiero, Ph.D. [fabio.ruggiero@unina.it]
Assistant: Riccardo Aliotta, M.Sc. [riccardo.aliotta@unina.it]
Assistant: Francesco Cufino, M.Sc. [francesco.cufino@unina.it]

[Updated 28/02/2024]

GRADING PROCEDURE (a.y. 2023-2024)

There are two ways to pass the FSR exam.

1. Take the assigned periodic homework in the given timeslots, the midterm, and present a final technical project.
2. Present a final complete technical report and undergo an oral discussion about the report and the class syllabus.

These ways hold for both inside and outside prescribed time students and students who followed this class years ago. For these last, notice that you completely adhere to this year's syllabus if you choose to follow one of the two paths indicated here. It is impossible to follow one of the paths below and study the syllabus related to the first year you attended the FSR class. Besides, if you already delivered some homework years ago, participating in one homework implies erasing all the history of all previous homework and midterm.

1) HOMEWORK + MIDTERM + FINAL TECHNICAL REPORT

Homework

N. HOMEWORK	RELEASED (MAX.DATE)	DEADLINE (NO EXTENSION)	RESULTS (FLEXIBLE)
1	19/03/24	29/03/24	12/04/24
2	16/04/24	26/04/24	10/05/24
3	07/05/24	17/05/24	31/05/24
4	28/05/24	07/06/24	21/06/24 (on demand)

Homework may be released before the indicated dates. All deadlines are fixed at 23:59 CET. Late submissions are not allowed. Results dates are more flexible.

Each homework is focused on one part of the course and consists of some small- or medium-size exercises to be solved. Some exercises may require a coding part.
For each homework, please, produce a detailed English report (maximum 15 pages, regular font) as a single PDF file. You may also upload the code. In this case, provide a unique ZIP file.
Be careful: it is not simply necessary to write the correct answer. You should explain the answer's reasoning while staying within the page limits.

Each homework is evaluated with the grades provided below. Missing a homework submission corresponds to an F grade.

GRADE	VOTE
A+	29-30
A	27-28
B	24-26
C	20-23

D	18-19
F	<18

The platform where submit the report and, eventually, the code is Moodle (https://mooduni.unina.it). You can find a Moodle tab within the Teams channel of the FSR course. If you cannot see anything in the tab, click the arrow near the sign “Moodle." Then, click on “Open in browser.” You must sign in with the UNINA credentials. Evaluation will also be given inside Moodle.

Midterm

The midterm will take place on April 29, 2024. It consists of 1-hour test in the classroom with the possibility to consult books and didactic material. You must bring with you an adequate number of sheets of paper, pen, and a calculator. The given test is like homework, but it will be carried out in the classroom, without code to produce.

It is prohibited the collaboration with other students during the midterm.

The midterm evaluation is carried out through the grades as before. If you do not show up at the midterm, for any reason, you take an F grade.

Remote participation to the midterm is allowed following the procedure indicated by the university for the remote exams.

Final technical project

The final technical project will be the opportunity to apply what the student learnt in the class and put it into action in an application of your choice.

Please, select the topic carefully. Each student should neither overestimate nor underestimate the work. It is expected that this project will take you a maximum of 3-4 weeks of work. It is possible to start working on the project during the class.

Developing novel algorithms/systems is okay, but reimplementing published work and/or applying a published algorithm to a model system from class is also great.

You will be evaluated based on whether the project forced understanding new concepts, not based on the novelty of the approach.

You can use the preferred coding language and simulation environment. In any case, help can be given by the instructor and the assistant on Matlab, C++, ROS, and Gazebo.

It is possible to extend a project carried out in previous classes. If this is the case, please, specify it in the report and highlight the novelty with respect to the previous report.

It is also possible to start working on something related to the M.Sc. thesis. The critical thing is to apply concepts you have seen in the FSR course; otherwise, the project will not be successful.

For the final technical, you should produce the following material.

- A final detailed report in English language (max. 25 pages). Try to avoid repeat theory studied in the class. Focus on your contributions and choices. The report should include the following.
    - An introduction of the project and why it has been selected as attractive.
    - The contribution made, also with respect to the class topics.
    - The obtained results with figures and plots to comment on the performance.
    - For team projects (see the policy below), please describe the contributions made by each team member.
    - For projects related to a project made in another class, please specify and highlight the novelty.
- The written code and any additional material.
- A video of the results.
- A presentation with slides to be shown during the examination. Tune the presentation to be maximum of 15 minutes. If you do the project in collaboration (see the policy below), each student's presentation should be 10 minutes, focusing on the part that each of you has developed chiefly.

To submit the project, please follow these steps.

1. Send an email to the instructor to agree on the final examination date. Please, do it at least 10 days before the day you have in mind for the final examination date.
2. Wait for the instructor's answer. After agreeing on a date, please, send all the material above to the instructor. You can add the instructor in the Git repository of the project if any.

Possible ideas of technical projects follow. The instructor and the assistants are open to discussion about further ideas and proposals. If you want to be sure you project fits the course's goals, feel free to contact us.

- Bicycle, autonomous car, quadrotor, or quadruped driving/flying/walking through a random forest/building/environment. Try to implement on a feedback controller that is independent form the map.
- Take all of parts of the complete technical report asked below for those not following this grading path.
- Perform a mobile manipulation task. The mobile robot is equipped with an arm and must pick up an object from one location to deliver it to another location. Obstacles can be avoided during the motion if you want to focus on planning techniques. You may also consider using feedback controllers for the robot motion and manipulation if you prefer focusing on the control part. Many examples can be found on the web with many robots to import into your preferred simulation environments.
- Robotarium (https://www.robotarium.gatech.edu) provides a remotely accessible swarm robotics research platform that is freely accessible to anyone. If you are interested in swarm planning and control (maybe, also linked to other courses), you may consider using this environment where you can upload your code, wait for the video and the results obtained from the real environment.
- A simple legged robot walking on rough terrain. The RABBIT model here (https://hal.archives-ouvertes.fr/hal-00794856/document) may be a good place where to start.

NOTICE: Starting from the 2023 class, presenting technical reports with unicycle/differential drive robots is NOT possible unless they are developed with the Robotarium. If you like wheeled robots, use another kinematics.

Grading

The final grade is given by

- 30% homework;
- 30% midterm;
- 40% final technical report.

There is not a final examination. Only a discussion about the final technical report to understand the carried-out work.

Suppose you are not satisfied with some evaluation regarding the homework, the midterm, and the final technical report. In that case, you can ask for a re-grade.
In this case, please, when asking for the final examination date, indicate the grade you want to discard and the reasons. After the presentation of the technical project, you will be asked an oral question for each topic included in the homework or midterm you wish to discard.

If you obtain

- three or more F grades during the homework,
  OR
- at least one F grade during the homework AND an F grade at the midterm,

you cannot pass FSR with this modality. You must follow the complete technical report + oral discussion path.

Collaboration policy

General rules:

- homework must be done individually;
- the final project can be done individually or in a small team (max 2 people, no exception!)

In general, collaboration is encouraged. You can talk with other students, the instructors, and the teaching assistants. However, the assignment is expected to be entirely on your own. This means that you can discuss with other students, but you must understand the concepts and produce them by your own assignment reports. The solution must be written by yourself. You must not see anyone else answer before writing yours.

Plagiarism is a violation of the honour code.

Anti-plagiarism software is used to detect plagiarism among all the received documents. Detected or suspected plagiarism, copying from other colleagues, copying from past years' assignments, and all similar actions will be penalized with an F grade or prevent taking the final examination with this path.

Generative AI

Generative AI is simply a tool, akin to any other, and you are at liberty to utilize it in your work. Should you choose to use it, citing it follows the same protocol as you would with any other reference.

2) COMPLETE TECHNICAL REPORT + ORAL DISCUSSION

General rules

- The assigned topic must be solved individually.
- A detailed English report (max 25 pages) must come along the produced code and a video showing the achieved results.
  - The source code, the PDF of the report, and the video must be upload on a private repository on GitHub.
    - When you are ready to "deliver" the project, the instructors must be added as a collaborator.
    - The README.md of the repository must contain a comprehensive description about how to download, compile, install, and run the solution. In this section, you must also add any reference to external dependencies needed to compile/run the project solution.
  - The link to GitHub repository must be sent to the instructor' email. The final oral examination day will be agreed with each instructor starting from one week later the submission of the technical report.
  - The report will undergo a plagiarism scan check.
  - The report should stress out the implemented choices and the motivations behind them. In particular, you should include the mathematical formulation and all the plots of interest to the resolution of the project.
- The projects are classified into three categories: *i)* wheeled robots; *ii)* aerial robots; *iii)* legged robots..
- The technical projects are assigned by the instructor based upon the students' surnames.
  - It is possible to ask the instructor, via email, for a change of the project category 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; etc.). 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 (Gazebo, CoppelliaSim, RaiSim, Simechanics, etc.), and so on. This does not hold for Project 4 where ROS and Gazebo is mandatory, but the code is already provided, and you must only add piece of it. There is instead a restriction on the employed libraries/commands: it is forbidden to use libraries/commands automatically solving the specific 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 instructor and the assistants are always at disposition to clarify each doubt.
- The oral discussion is made by a presentation of the report (10-15 minutes) plus three questions randomly covering all the topics of the syllabus.

TECHNICAL PROJECTS ASSIGNMENT

- Surnames A-C: Project 1
- Surnames D-G: Project 2
- Surnames H-N: Project 3
- Surnames O-Z: Project 4

TECHNICAL PROJECTS DESCRIPTION – PROJECT 1

Robot: Differential-drive robot
Characteristics: Perform 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. It is possible to use already existing imported models.
Scenario: The goal of this project is to develop a control system for a wheeled mobile robot used in a logistic environment. The environment is composed of three rooms: a warehouse and two destination rooms. In this scenario, the robot must transport an object from the warehouse to a destination room. The map and the coordinates of each room are known. During the whole task, the robot must be able to localize itself into the map without using precise information provided by the simulator (*i.e.*, odometry and additional sensors must be used to solve localization and/or simultaneous localization and mapping problem).
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 (FSR): Free to be chosen among the tracking and/or the regulation controllers illustrated during the course. Please, put attention on the fact that the feedback is given by the estimated pose extracted by the odometry/localization technique. You do not have to feed back to the controller the pose retrieved from the simulator!

TECHNICAL PROJECTS DESCRIPTION – PROJECT 2

Robot: Differential-drive robot
Characteristics: Perform 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. It is possible to use already existing imported models.
Scenario: The goal of this project is to develop a control system for a wheeled mobile robot used in an exploration task. The environment is composed of four main locations: a starting location and three other locations to be explored. In this scenario, the robot must travel towards the locations to be explored to find a given object. The coordinates of the locations are known. During the whole task, the robot must be able to localize itself into the map without using precise information provided by the simulator (*i.e.*, odometry and additional sensors must be used to solve localization and/or simultaneous localization and mapping problem).
Planning (FSR): Planning must be carried out with an online version of the artificial potential method. The force field is seen as acceleration vector. Use random solutions to avoid possible local minima.
Control (FSR): Free to be chosen among the tracking and/or the regulation controllers illustrated during the course. Please, put attention on the fact that the feedback is given by the estimated pose given by the odometry/localization technique. You do not have to feed back to the controller the pose retrieved from the simulator!

TECHNICAL PROJECTS DESCRIPTION – PROJECT 3

Robot: Quadrotor.
Characteristics: 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. It is possible to use already existing imported models. It is possible to use the RotorS library.
Scenario: The goal of this project is to develop a control system for an aerial robot employed in a navigation task. This robot must navigate over a set of predefined waypoints avoiding obstacles. Obstacles can be classified into two types: solid obstacles and empty objects. In the first case, the

obstacle must be avoided flying around it. In the second case, the robot must pass through it (you can consider it as a window). The obstacles must be placed at different altitudes!
During the whole task, the robot must be able to localize itself into the map using precise information provided by the simulator.
Mission: The map of the environment known. The robot must navigate all the waypoints passing through the empty obstacles. The other obstacles must be avoided. The resulting path should be non-trivial.
Planning: You can use any known planning technique, also that you find on the web.
Control (FSR): Free to be chosen among the controllers illustrated during the course. The controller must also deal with uncertainty about the knowledge of the dynamic parameter of the robots. To this end, you must implement an estimator to compensate for the related effects.
In detail, the mass in the controller is underestimated by 10%. Besides, an external force of 1N is continuously acting along the x-direction of the world inertia frame. Notice that the controller can be turned off during take-off and landing phases: in these cases, planning and feedforward techniques can be employed.

TECHNICAL PROJECTS DESCRIPTION – PROJECT 4

Robot: Quadruped.
Characteristics (RL/FSR):
The quadruped is the DogBot (https://github.com/rl2021/dogbot).
For kinematics and dynamics, it is necessary to use iDynTree library (https://robotology.github.io/idyntree/index.html) that must be installed on your PC.
For gait planning it must be used the Towr library. The suggested gait to be employed is the *trot*. To solve the optimization problem within the controller, the C++ AlgLib library must be used. Towr, AlgLib, and a main skeleton of the program is provided as a ROS package (https://github.com/rl2021/popt). To start the simulation, you must use the *launchfile* included in this package through the following command

$ roslauch popt dogbot.launch

Additional documentation is available within the provided code. The sections with TODO are obviously at your charge.
Scenario: This project aims to develop a control system for a legged robot used in an exploration scenario. The environment comprises four main locations: a starting location and three other rooms to be explored. In this scenario, the robot must travel towards the locations to be explored to find a given object. The coordinates of the locations are known. During the whole task, the robot must be able to localize into the map using direct information provided by the simulator.
Mission: The map of the environment is known. The robot must visit in sequence the three locations to be explored. *S*.
Planning: You can use any known planning technique, also that you find on the web.
Control (FSR): Use the design based on the optimization control problem illustrated during the course. 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%. To this end, you must implement an estimator to compensate for the related effects.