Project
Robot Motion Planning and Control

Due date: Friday, **April 3rd, 2020**, before the end of the day

Course: ME47035 Robot Motion Planning and Control, TU Delft
Contact (setup/report): Francisco Martínez Claramunt, email: F.MartinezClaramunt@student.tudelft.nl
Contact (methods): Álvaro Serra Gómez, email: A.SerraGomez@tudelft.nl

Alternative contacts:
Contact (MAVs): Hai Zhu, email: H.Zhu@tudelft.nl
Contact (cars/RRT/MPC): Bruno Brito, email: Bruno.deBrito@tudelft.nl
Contact (algorithms): Javier Alonso-Mora, email: j.alonsomora@tudelft.nl

1 Introduction

In this course you will learn several methods for motion planning and control and be able to answer questions such as: How does an autonomous car safely navigate in a city? How does a micro air vehicle find its way in a building? How to coordinate a fleet of self-driving cars or warehouse robots? Or, how does a mobile manipulator interact with its environment?

In this project you will choose a robotic application of your interest. Then, you will describe the kinematics of the robot, implement a path/motion planner, and apply it to solve a navigation task. By doing this, you will solve a challenging problem and practice at least one of the methods learned in this course.

In the last lecture you will present your work to the class and learn from the experience of other groups.

2 Learning objectives

The goal of the project is to describe, implement and validate a motion planner for a mobile robot.

Within the scope of your project, the learning objectives are:

- Identify the robotic system and the associated mathematical model for its kinematics, dynamics and motion equation.
- Identify the workspace and configuration space in which the robot operates.
- Describe the chosen algorithm for motion planning and control of the mobile robot. You will also justify your choice.
- Design, implement and evaluate the chosen motion planner to solve a motion-planning task and navigate a mobile robot.
- Describe the latest developments in autonomous mobile robots. You will do this by comparing your method to the state of the art in robotics.

3 Instructions

The project consists of several deliverables and deadlines described in the following.

The project is to be performed in groups of four students, with a single submission per group. Your group must be registered in Brightspace by February 19th, 2020. Register with your group in the next available group slot corresponding to the topic of your choice (see next section for more information). If you do not have a group, please join the last available group slot of your preferred topic. Groups with only 1, 2 or 3 students will be split or merged at the end with other semi-full groups. Only groups of 4 students are allowed.

The project is compulsory. If you are not registered by the deadline, you are assumed not to follow the course.

Submit by Brightspace **and** leave a printed copy of the report in the mailbox of J. Alonso Mora (In front of 3mE F-2-400).

4 Topic

For the project you must select ONE motion planner and ONE robot morphology from the list below. You may propose a different topic (related to the course) to be discussed with the instructor. The topic must be chosen by February 19th.

- Robot morphology: Choose ONE robot morphology from: car (with or without trailer), quadrotor.
- Motion planner: Choose ONE motion planner from: motion primitives (or lattice), RRT (or its variants), model predictive control.
- Map: Define an environment, which may contain static and/or moving obstacles.

You may choose your preferred simulation environment (e.g., Matlab, Python, V-REP, ROS) and may use state of the art solver packages that are freely available online (in your report you must clearly state the main software packages that you rely on).

5 Deliverables

Three deliverables are required at the end of the project, a report, a presentation and the source files. The weight of each part is given as percentage of the total.

A group grade will be computed, which can be adjusted with an individual performance index.

5.1 Report

(80%) The report shall have a maximum of four pages in IEEE conference format (A4, 11pt), including figures. References do not count towards the page limit. The first page must include the title of the project and the student names and netIDs.

Your report shall consist of the following parts:

1. (10%) Introduction: brief description of the project and brief comparison to state of the art (with references to publications used in the project or relevant to the project). Brief description of the task to be achieved by the robot. Recommended length: 1/2 page.
2. (15%) Robot model: Description and equations (with brief derivation) of the robot kinematic/dynamic model and the equations of motion (10%). Describe the workspace and configuration space (5%). Recommended length: 1/2-1 page.

3. (15%) Motion planning: Description and algorithm for the chosen motion planner, with the required adjustments for the project. Recommended length: 1/2-1 page.
4. (20%) Results: Description of the simulation setup. Description of results, including figures and plots. Note: You may use the same figures and results for the presentation. You may include a link to a video showing the performance of the system. Recommended length: 1 page.
5. (20%) Discussion: Both theoretical (e.g., is the method optimal? complete? what is the complexity?) and practical (e.g., did it perform well, would another planner have performed better? what are the limitations of the method?) (15%). Recommendations for extensions with how the planners could be improved (5%). Recommended length: 1/2-1 page.
6. Bibliography: References used in the project or in the comparison to state of the art. Recommended length: 1/4-1/2 page.

Note: Up to 10% additional bonus will be given to innovative/challenging projects and to groups that implement their own algorithms, up to a maximum grade of 100%.

5.2 Presentation

(20%) A three minutes presentation, with the following format (template):

1. Title slide, project number + title + members
2. (5%) One slide with project brief description of robot and planner
3. (10%) One slide: video with simulation results
4. (5%) One slide: discussion

5.3 Source files

You must provide a link to download all the source code developed in the project (model, controller, planner, simulation) so that we can run it if needed.

5.4 Group peer review

You will evaluate your own work and that of your group members following the provided evaluation form. For each of you, the evaluations by all team members will be combined in an individual performance factor [IPF] with range 0.8-1.2. The average IPF of all the students in a group will be 1.0. If the variations between students in a group are small, then all students in the group will get an IPF of 1.0.

6 Deadlines

The following deadlines are observed (for exact dates refer to the course syllabus in Brightspace).

- **Wednesday, February 19th, 2020**: Register groups (or indicate participation in project) and choose topic.
- **Monday, March 2nd, 2020**: Submit 1-2 page report with the robot model and equations of motion. May include the controller. [OPTIONAL AND NOT GRADED]
- **Tuesday, March 10th, 2020**: Feedback will be provided on the submitted reports.
- **Tuesday, March 31st, 2020**: presentations.
- **Friday, 3rd April, 2020**: report.

7 Grading

Grading of the report and the presentation will follow a standard grading rubric. The weight of each part is specified in the Deliverables section. This will lead to a group grade [GP].

The individual grade will then be given by [GP]×[IPF] and converted to a 0-10 scale with a maximum of 10 and rounded at one decimal point.

8 Tips

To record the video you may use screen capture in your computer.

9 Links to some software packages available online

- Robotics Toolbox: `http://petercorke.com/wordpress/toolboxes/robotics-toolbox`
- OMPL: `http://ompl.kavrakilab.org/`
- MPC solver Acado: `http://acado.github.io/`
- MPC solver Forces PRO: `https://www.embotech.com/forces-pro/`
- Optimal control solver Casadi: `https://web.casadi.org/`
- MAV simulator in ROS: `https://github.com/ethz-asl/rotors_simulator`
- Fleet management: `https://github.com/idsc-frazzoli/amod`