Control of Mobile Robots
Project work - Group 6 (2 students)
Prof. Luca Bascetta

Robot modelling and simulation

Consider a rear-wheel drive car modelled as a bicycle robot, characterised by the following mechanical parameters:

$$m = 140\,\mathrm{kg} \qquad J_z = 2500\,\mathrm{kg\ m}^2 \qquad a = 1.1\,\mathrm{m} \qquad b = 1.6\,\mathrm{m}$$
$$C_f = 85000\,\mathrm{N/rad} \qquad C_r = 70000\,\mathrm{N/rad} \qquad \mu = 0.65$$

Create:

- a Simulink diagram representing a kinematic simulator of the car, modelled as a bicycle, whose interfaces are a Twist message as input (where the rotational velocity represents the steering angle) and an Odometry message as output, and generate the code for the corresponding ROS node;
- a Simulink diagram representing a dynamic simulator of the car, able to consider either a linear or a non-linear tire-ground interaction model, whose interfaces are a Twist message as input (where the rotational velocity represents the steering angle) and an Odometry message as output, and generate the code for the corresponding ROS node.

Clearly explain how the node rate and the message publishing rate are selected.

Planning

Write a Matlab script that plans a trajectory using sPRM and RRT in a randomly generated maze environment, from the entrance to the exit of the maze.
To generate the maze use the application provided at *www.mazegenerator.net*, and consider hexagonal cells with size 20×20.

Clearly explain how the planner parameters are selected, and compare the plans obtained with the two planners for different values of the parameters, considering the quality of the plan in terms of length, number of nodes, computation time, smoothness (variation of the curvature with respect to a variation of the length), etc.

Control

Create a Simulink diagram implementing a trajectory tracking controller composed of a feedback linearising law, designed on the bicycle kinematic model, and a proportional trajectory tracking controller with velocity feedforward, and generate the code for the corresponding ROS node. The interfaces of this node are a Twist message as output, and an Odometry and Path message as inputs.

Clearly explain how the controller parameters are selected.

Validate the tracking controller following the planned path and showing how the position of point P, i.e., the point used to compute the linearising feedback, influences the feasibility of the path planned with sPRM and RRT in the presence of close obstacles.

Instructions and evaluation

Project work can be implemented directly in C++, or generating C++ code using Matlab/Simulink.

When the project is completed, create a compress file including:

- Matlab/Simulink or C++ files that allow to solve the proposed problem, including launch files and script files to plot the results;
- a *README.txt* file explaining, for each of the parts of the project, what is (are) the launch file(s) that have to be used to generate the solution, and what is (are) the script file(s) to plot the results;
- the Power Point presentation that will be used to discuss the results at the oral exam.

and use as file name "2020-CMR-projectwork-group$X$.zip", where X is the number of the group.

Send the compress file by email at least three working days before the date of the oral exam.

Remember that, as far as evaluation is concerned, the most important aspect is that every single design choice is suitably explained and motivated. In case some parts are not working properly or not in every situation, do not worry, but remember to clearly analyse the limitations of your approach and explain the reasons of the failure.