MTRN4110 – Project 4

Introduction:

Project 4 (P4) is the continuation of Projects 1 and 2. In P4 we develop additional modules, to give the platform full perception, and the capability to perform planning.
In this project, we solve the following capabilities:

1) Localizing the platform, based on IMU and 3D camera measurement, and on a provided Navigation Map.

2) Occupancy Grid (OG) module, for modeling the presence of obstacle; and to be used for defining the cost of traversing the terrain (called local cost, LC), for finally obtaining an optimal path, for reaching X_{goal} , the desired destination (the goal).

The project is composed by four parts.

Part 1 (can be solved working in teams of up to four, 4, members)

Implement a module for localizing the platform. Details and recommendations are given in document "InformationForProject4_Part1.pdf". This part of the project can be solved in teams, of up to 4 members.

Part 2:

Implement a module for generating, dynamically, an Occupancy Grid (OG). The processing must include the *set* and *erase* operations. In addition to maintaining the OG (i.e. numerically, in a matrix), you will show its content through proper visualization. Details about the recommended visualization are described in the section about recommendations for solving this project.
This OG must be updated at a rate of, at least, once a second (1Hz). Higher rates are also adequate, but those are not required in this project, in order to reduce processing cost. Grid resolution is defined by the students, although cells of 5 cm by 5 cm is recommended.

Part 3:

3.1) Generate a Local Cost function (LC), numerically represented through a matrix, based on the OG obtained in Part 1. The LC must also consider inflating the obstacles (regions inferred to be obstacles, in the OG); this is done in order to deal with the size of the platform (which is bigger than one cell).
It is assumed that a radius of about 15cm is an adequate value for the size of the platform; however, this parameter should be modifiable, in your program.
The values of the LC must be scaled according to the convention required by the planner (to be required in subsequent parts of the project).
3.2) Define a "Virtual Obstacle". Although we do not see it with our sensors, we need to define certain region of exclusion, i.e. a region which the platform is not allowed to invade. The region has a rectangular shape. Propose a virtual obstacle and add it to the OG or directly to the LC. The virtual obstacle should be visible in the OG visualization or in the LC's one. You may define multiple virtual obstacles for testing the planning process, although it is not required in this project.

Part 4 (can be solved working in teams of up to four, 4, members, same team created for part 1):

a) Obtain an optimal path (in the sense of "shortest distance"), from the platform's current position to the currently specified goal. For that, you can use some of the planners (Dijsktra or A*), which are provided in the Matlab community. (Note: You should NOT use the MEX file planner which was provided by the Lecturer, years ago, to MTRN4110 and theses students.)
This planning process must be performed at a rate of no less than 0.5 Hz. A higher rate, e.g. 1Hz, is also recommendable, but not necessary.
In this part of the project, the desired destination point can be defined in a variable, at the start of the program.

b) Allow users to specify the destination, through some basic GUI capabilities (e.g. by clicking certain mouse's button at the desired point, in the OG figure). By definition, the current desired destination is the last destination specified by the user. This means that the destination can be modified at any time during the experiment.

Part 5:
The previous parts are solved using the same data you used in P1 and P2. In this part, you are required to implement all the previous parts, for operating in Real Time. You will receive the data via UDP. This would allow you to try it with the real robot, on week 13.
Example source code and a playback module are provided for testing these parts.

Recommendations for the visualization of results

Your program will show the area of operation, the provided map (landmarks), and the position and heading of the platform. You may show this separately, or jointly with the Occupancy Grid information.
For the OG: The OG will be shown through an image, which will be refreshed dynamically, via Matlab's *set* function. The image will clearly show the regions that are inferred to be occupied, clear and unknown. A possible color convention may be the following one: unknown cells = black, clear cells = dark gray, occupied cells= range of gray intensities from light grays to white. You may use other color conventions, not necessarily monochrome.
In the same figure, you will show the position of the known navigation poles (the landmarks), the current 2D position and heading of the platform, and the currently detected OOIs (expressed in the global coordinate frame). The scale of each axis, (X, Y), will be expressed in meters or centimeters (but not in "cells").
The refreshing rate for this image should not be lower than 0.5 Hz (once every 2 seconds). A rate of 1Hz is recommended but not strictly required.
For part 4, the current optimal path must be also shown, in the same OG's figure. Its updating rate should be not less than 1Hz.

Relevance of Project4's parts: Part 1: 20%; Part 2: 25%; Part 3: 25%; Part 4: 15%; Part 5: 15%

Deadlines: Demonstration: On week 13 (during your session time). Submission of programs: The day of your demonstration.

Quiz: There will be a Quiz, in week 13, in your lab session, before demonstration time. It will run from minute 20 till minute 35. You are expected to attend it, even if you give your demonstration in a different week or day. If you are not able to attend, contact the lecturer or apply for Special Consideration.

Alternative project for students who prefer working **individually**.
Solving Part 1 and 2, working individually.
Relevance of parts: part 1: 60%; part 2: 40%.

Questions: Via Moodle's Forum or by asking the lecturer by email (j.guivant@unsw.edu.au)
