Final project suggestions

Here are some suggestions for final projects. Most are suitable for 2 or 3 person projects, though of course I expect a 3 person project to have a larger scope than a 2 person project. It is fine for multiple teams in the class to be working on the same or similar projects.

Some random thoughts on the projects:

- Being able to display the results of your project is important, but there shouldn't be too much emphasis on user interface. (This is a class on robotics, not on HCI!)
- If appropriate, you can consider using input from the camera. I can provide code for detecting colored markers in an image. However, I will generally not encourage it for all projects.

Integrated motion planning and execution

Using a given map, plan a path for the robot from its current location to a goal location and then execute this path. You should monitor the robot's execution and detect when the robot is either too far from the nominal path or when the world differs too much from the map. You should then replan the path. In order to do this, you will need to incorporate either localization (so you know where the robot is in relation to the original goal), mapping (so you know how to correct the map and plan a new path), or perhaps both.

Motion planning and execution with time-optimal paths/trajectories

In Assignment 1, you implemented a motion planning algorithm that generated piecewise linear paths. For a differential drive robot to execute these paths, it must stop at the end of each segment and turn before starting on the next segment. In general this is not the fastest way to get from one point to another. (The fastest path is not always the shortest path!)

There has been quite a bit of work on generating time-optimal paths and trajectories for differential drive mobile robots. (The former is easier than the latter!) You could start by implementing path generation and execution for time-optimal paths and then extend this either by devising an algorithm to plan these paths in the presence of obstacles or by figuring out how to use feedback control to track these paths. The more ambitious can think about tackling time-optimal trajectory generation.

Concurrent Mapping and localization

You won't solve the general problem of metric mapping and localization in the next month, but you could tackle some simplified version of that problem. One possibility would be to use the camera to identify landmarks (or perhaps just markers) that can be used to completely or partially localize the robot (possibly using the Kalman filter!) With this localization aid, create a map of the lab with obstacles (let's assume obstacles are empty boxes). This could be extended by incorporating your Assignment 1 motion planner to plan paths using the generated map.

Topological mapping

Develop a "corridor following" behavior and use it to create a topological map of an unknown area. This map should be annotated at least with the distance between nodes. Once learned, use this map for planning a path for the robot from the robot's current configuration to another point in the area.

Some questions for this project are: How will the robot follow corridors? What assumptions are you making about corridors? How will nodes in the topological graph be recognized?

Although you can start this project in the lab using cardboard-wall corridors, you should plan on being able to navigate the first floor of Amos Eaton.

A three person version of this project should add something to its scope. One possibility would be to recognize doors in the corridor and then being able to navigate to a particular office.

The kidnapped robot

The "kidnapped robot" problem is when the robot is suddenly transported to some location in its map, and then the robot must globally localize itself. For this project, you could provide the robot with a topological or metric map of the first floor of Amos Eaton, move it to some location, turn it on, and have it find its way back to the lab.

Extensions to this project could include building the map or having the robot find its way to its "docking station." Like the previous project, this could be started in the lab using cardboard walls, but you should plan on being able to navigate the first floor of Amos Eaton.

Reactive robotics

Design a collection of reactive behaviors to do some useful or interesting task. How about a "clean up" task: push (empty) boxes strewn on the lab floor to one side of the lab.