Machine Learning Engineer Nanodegree
Capstone Proposal - Robot Motion Planning

Tien Pham
July 4, 2018

Plot and Navigate a Virtual Maze

I. Definition

1. Project Overview

This project was inspired from Micromouse competitions originated in the 1970s, wherein a robot mouse is given the task of running multiple time in a given maze and plotting the best path from a corner of the maze to its center. In this project, a simplified model of the world is provided along with specifications for the maze and robot. The goal of this project is to obtain the fastest times possible in a series of test mazes by controlling the navigation of a virtual robot in a virtual maze.

2. Problem Statement

On any given maze, the robot must complete two runs, each trial run tries to solve a different problem. During the first run, the robot is allowed to freely roam, explore, and analyze the maze to learn and map the structure of the maze, and find all possible paths to the goal. To finish the first run, it must enter the goal room during its exploration. After entering the goal room, the robot is free to continue exploring the maze and may choose to end its exploration at any time. The robot is then moved back to the starting position and orientation for its second run. Its objective now is to go from the start position to the goal room following an optimal path in the fastest time.

3. Metrics

The most basic form of the performance measurement is a maximum of 1000 steps to complete both runs. Both exploration trial run and optimization trial run impact on quantifying the performance of the benchmark model. The following metric will measure the performance score for the benchmark model:

Score = [no. of steps in trial 2] + [no. of steps in trial 1 / 30]

II. Analysis (2-4)

1. Data Exploration

The information of mazes is provided in a text file. On the first line of the text file is a number describing the number of squares on each dimension of the maze n, n is even and in a range [12, 16]. On the following n lines, there will be n comma-delimited numbers describing which edges of the square are open to movement. Each number represents a four-bit binary that corresponds four-side, in this order: up, right, down, left. A bit value of 1 if there is no wall, and a bit value of 0 if an edge is closed (walled).

The starting point and the goal room are "known" to the mouse before it starts. The starting point is always in a left corner and is always surrounded by three walls. The goal room is four center blocks of the maze.

The robot has three obstacle sensors mounted at the front, right and left which detect the number of open squares in the direction of the sensor. These sensors detect the number of open squares in the direction of the corresponding sensor, then store in the form of a list of three numbers.

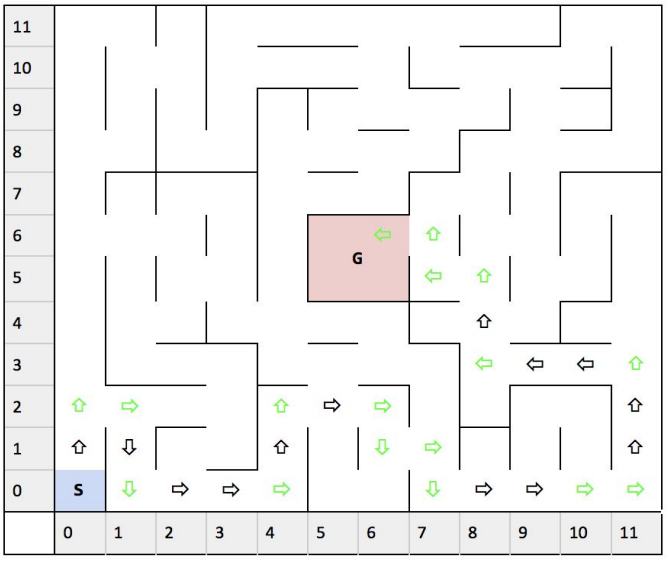
A 2-digit pair will define the location of the robot mouse as the coordinate [x, y]. The robot decides next location with two values indicating the rotation and movement on that timestep. Rotation is an integer taking one of three values: 0, 90, or -90, indicating a no rotation, clockwise, or counterclockwise. Movement is also an integer in the range [-3, 3]. The robot will attempt to move that many squares forward (positive) or backward (negative), stopping movement if it encounters a wall.

2. Exploratory Visualization

The following figure is described in the layout of ***Test Maze 1*** with wall description number. In this maze visualization, there are some squares with red numbers, which are dead ends the robot should recognize and avoid.

11	6	12	4	6	10	10	14	14	10	8	6	12
10	5	7	13	7	10	10	13	3	14	14	9	5
9	5	5	5	5	4	6	11	14	9	7	8	5
8	7	9	3	9	7	11	14	9	6	15	10	9
7	5	4	6	12	7	10	9	6	13	5	6	12
6	7	15	13	5	5	6	14	13	7	13	5	5
5	5	5	7	13	5	3	9	3	13	7	9	5
4	5	7	9	3	15	10	12	2	15	9	2	13
3	5	3	10	12	3	14	11	12	7	10	10	13
2	7	14	10	13	6	15	12	5	1	6	12	5
1	5	5	6	9	5	5	7	13	4	5	5	5
0	1	3	11	10	9	3	9	3	11	11	11	9
	0	1	2	3	4	5	6	7	8	9	10	11

The start location is [0, 0] and the goal area wherein the center of the maze is marked with a light red color. In this maze, the shortest path from the start to the goal is indicated the arrows. The robot should move 30 single steps to reach the goal. However, the robot is allowed to take a maximum of three steps in one movement. So, the number of steps decreases to 17 steps, the best path to the goal is marked with green arrows.



3. Algorithms and Techniques

In this section, you will need to discuss the algorithms and techniques you intend to use for solving the problem. You should justify the use of each one based on the characteristics of the problem and the problem domain. Questions to ask yourself when writing this section:

- *Are the algorithms you will use, including any default variables/parameters in the project clearly defined?*
- *Are the techniques to be used thoroughly discussed and justified?*
- *Is it made clear how the input data or datasets will be handled by the algorithms and techniques chosen?*

4. Benchmark

In this section, you will need to provide a clearly defined benchmark result or threshold for comparing across performances obtained by your solution. The reasoning behind the benchmark (in the case where it is not an established result) should be discussed. Questions to ask yourself when writing this section:

- *Has some result or value been provided that acts as a benchmark for measuring performance?*
- *Is it clear how this result or value was obtained (whether by data or by hypothesis)?*

III. Methodology (3-5)

1. Data Preprocessing

In this section, all of your preprocessing steps will need to be clearly documented, if any were necessary. From the previous section, any of the abnormalities or characteristics that you identified about the dataset will be addressed and corrected here. Questions to ask yourself when writing this section:

- *If the algorithms chosen require preprocessing steps like feature selection or feature transformations, have they been properly documented?*
- *Based on the Data Exploration section, if there were abnormalities or characteristics that needed to be addressed, have they been properly corrected?*
- *If no preprocessing is needed, has it been made clear why?*

2. Implementation

In this section, the process for which metrics, algorithms, and techniques that you implemented for the given data will need to be clearly documented. It should be abundantly clear how the implementation was carried out, and discussion should be made regarding any complications that occurred during this process. Questions to ask yourself when writing this section:

- *Is it made clear how the algorithms and techniques were implemented with the given datasets or input data?*
- *Were there any complications with the original metrics or techniques that required changing prior to acquiring a solution?*
- *Was there any part of the coding process (e.g., writing complicated functions) that should be documented?*

3. Refinement

In this section, you will need to discuss the process of improvement you made upon the algorithms and techniques you used in your implementation. For example, adjusting parameters for certain models to acquire improved solutions would fall under the refinement category. Your initial and final solutions should be reported, as well as any significant intermediate results as necessary. Questions to ask yourself when writing this section:

- *Has an initial solution been found and clearly reported?*
- *Is the process of improvement clearly documented, such as what techniques were used?*
- *Are intermediate and final solutions clearly reported as the process is improved?*

IV. Results (2-3)

1. Model Evaluation and Validation

In this section, the final model and any supporting qualities should be evaluated in detail. It should be clear how the final model was derived and why this model was chosen. In addition, some type of analysis should be used to validate the robustness of this model and its solution, such as manipulating the input data or environment to see how the model's solution is affected (this is called sensitivity analysis). Questions to ask yourself when writing this section:

- *Is the final model reasonable and aligning with solution expectations? Are the final parameters of the model appropriate?*
- *Has the final model been tested with various inputs to evaluate whether the model generalizes well to unseen data?*
- *Is the model robust enough for the problem? Do small perturbations (changes) in training data or the input space greatly affect the results?*
- *Can results found from the model be trusted?*

2. Justification

In this section, your model's final solution and its results should be compared to the benchmark you established earlier in the project using some type of statistical analysis. You should also justify whether these results and the solution are significant enough to have solved the problem posed in the project. Questions to ask yourself when writing this section:

- *Are the final results found stronger than the benchmark result reported earlier?*
- *Have you thoroughly analyzed and discussed the final solution?*
- *Is the final solution significant enough to have solved the problem?*

V. Conclusion (1-2)

1. Free-Form Visualization

In this section, you will need to provide some form of visualization that emphasizes an important quality about the project. It is much more free-form, but should reasonably support a significant result or characteristic about the problem that you want to discuss. Questions to ask yourself when writing this section:

- *Have you visualized a relevant or important quality about the problem, dataset, input data, or results?*
- *Is the visualization thoroughly analyzed and discussed?*
- *If a plot is provided, are the axes, title, and datum clearly defined?*

2. Reflection

In this section, you will summarize the entire end-to-end problem solution and discuss one or two particular aspects of the project you found interesting or difficult. You are expected to reflect on the project as a whole to show that you have a firm understanding of the entire process employed in your work. Questions to ask yourself when writing this section:

- *Have you thoroughly summarized the entire process you used for this project?*
- *Were there any interesting aspects of the project?*
- *Were there any difficult aspects of the project?*

- *Does the final model and solution fit your expectations for the problem, and should it be used in a general setting to solve these types of problems?*

3. Improvement

In this section, you will need to provide discussion as to how one aspect of the implementation you designed could be improved. As an example, consider ways your implementation can be made more general, and what would need to be modified. You do not need to make this improvement, but the potential solutions resulting from these changes are considered and compared/contrasted to your current solution. Questions to ask yourself when writing this section:

- *Are there further improvements that could be made on the algorithms or techniques you used in this project?*
- *Were there algorithms or techniques you researched that you did not know how to implement, but would consider using if you knew how?*
- *If you used your final solution as the new benchmark, do you think an even better solution exists?*