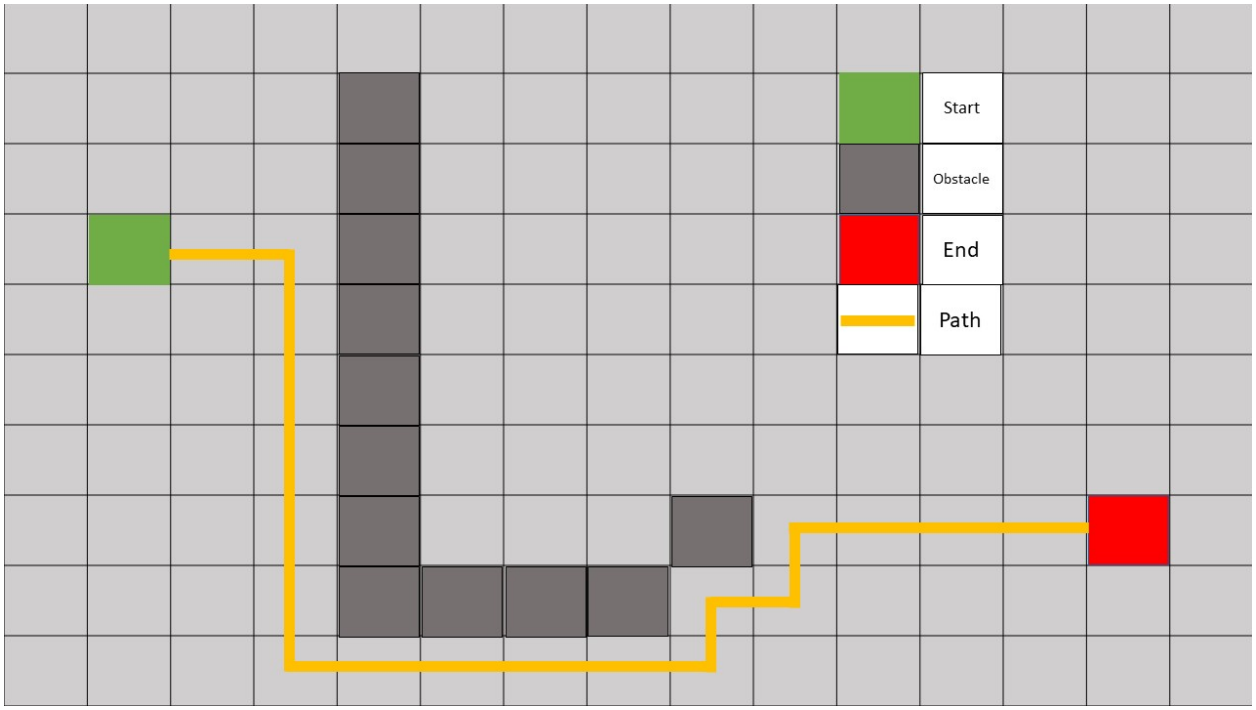
Ashish Roongta, Bryan Zhao, Pranav Narahari, Yayati Jadhav **(BYPA)**
Engineering Computation, 24-780:B
November 4, 2018

Team Project – Conceptual Design

Product Name: **Adaptive Motion Planning Simulator for Mobile Robotics (AMPS)**

Product Description: This software product is a simulation tool that allows researchers, students and hobbyists to experiment with various classical motion planning algorithms. Specifically, this tool aims to generate a virtual environment in which the user can define a starting point, objective goal, and any number of obstacles. Based on the motion planning algorithm (e.g. Dijkstra's algorithm, A* Search, D* Search, RRT) selected by the user, an optimized path between the two waypoints will be calculated, and the animated mobile robot will then traverse the trajectory. The optimized path will be highlighted and change according to the user input, and subsequently, the simulator will save the coordinates of the path in a separate datafile for the user's convenience.



Product Identification:

What are the basic functions of the product?

The product is a software package capable of simulation and visualization of multiple path planning algorithms. The motion planning algorithms will generate optimized paths for different environments, given the start and end points as well as obstacles. The user will also be able to add

obstacles in the path of the robot while it is in motion in real-time, and the robot will accordingly alter the optimized path. The product will also output optimized paths as spatial coordinates that can be implemented in any other tools for further work.

What are the special features of the product?

1. The product is aimed at developing a platform for visualization and simulation of multiple motion planning algorithms.
2. Output the optimal path from start point to end point for each environment and set of obstacles.
3. Real time path adaptation with dynamically changing obstacles.
4. The product will output the spatial coordinates of the optimal path to a file that can be imported in any subsidiary motion planning tool or a real mobile robot.

What are the performance targets?

- Allow users to intuitively run a simulator of mobile robot in a 2D environment.
- The paths should be mapped out in the GUI, and highlighted to allow intuitive understanding how the robot interacts with the environment and moves from a start to end position, manipulating trajectory around obstacles in place.
- The program will also save the planned trajectory and output the vertices or points along the path to a file for the user's convenience.

Market Identification:

What is the target user group? Researchers, students and hobbyists who have little or no prior experience in path planning and want to build a robot/system that needs to follow a path or simply someone wish to have an open-source visualization tool of classic motion planning algorithms.

How large is the user group?

The target user group for this product includes everyone who are new to the field of motion/path planning. Also, user's need no previous experience in path planning, thus increasing the user target group to even enthusiasts and hobbyists who are interested in building a robot/system that needs to follow a path or visualize and compare multiple motion planning algorithms.

What are the competing products? Currently, the common robotic simulators include Gazebo and V-REP, but these utilize 3D environments and have steep learning curves for first-time users. These competing products require great computational power (since the rendered graphics are very high quality), take some time to install and get familiar with the packages, and are quite confusing to navigate. Our software product aims to utilize a simple pipeline that has a more specific application (classical motion planning and animation) that is much more user-friendly and intuitive.

System Description:

What are the major functional components in the product? (Assign each of the components to one of the team members.) The major functional components in the product are listed below and assigned to each of the team members.

Ashish Roongta - Dijkstra's Algorithm, setting up the visualization environment, animating user-defined grid, assembly/integration leader.

Bryan Zhao - A* Search Algorithm, visualizing simple obstacles and path, animating mobile robot.

Pranav Narahari - RRT, developing features to allow users to select different algorithms (i.e. some UI design), presentation and report.

Yayati Jadhav - D* Search Algorithm, developing features to output and save data for different paths (file I/O), software integration of all features.

How should the functional components be assembled together?

Different classes will be formulated for visualization, each motion planning algorithm and file I/O. The classes and member function definitions will be compiled to a header file and a .ccpp file in Visual Studio. We will be using GitHub for version control of all individual scripts, and assigning one person (primarily Ashish Roongta) for integrating the software and running tests to validate the functionality of the programs.