ECE 417
Progress Report 1
Evan Desmond and Louis Laurita
19 November, 2023

1. The team consists of Evan Desmond and Louis Laurita. We are both taking the 498 undergraduate variety of this course.

2. Our project consists of two major goals:
   (i) Finish implementing the assignment given by Midterm 1.
   (ii) Extend the scope of Midterm 1 to include avoidance of both the static obstacles given by the Aruco tags but also moving obstacles such as another jetbot or moving Aruco tag.

   This topic was chosen because the team members are quite familiar with the content and scope of Midterm 1, and want to properly accomplish the assignment. An already solid understanding of the initial project leaves room for these topics to be extended into more complicated cases, such as considerations for dynamic objects as opposed to static objects. Additionally the team is already familiar with ROS nodes, jetbot motor control, and Aruco tag detection, so there is some confidence that nothing will be encountered which might completely halt progress on the project.

3. Why is this project important?

- While static environments do exist often the environments that mobile robotics operate in are changing and moving and the robots need to be able to account for this. For example the Kiwibots on campus navigating the sidewalks with people walking on them would not be possible without dynamic object detection and avoidance.
- Dijkstra is often cited as a simple and effective algorithm for dynamic object avoidance, and this project will hopefully add support to this claim. It will also be an opportunity to acknowledge the limitations or shortcomings of using Dijkstra's algorithm for this type of problem.

4. Related work:
   (i) "Application of Dijkstra algorithm in robot path-planning" →
   https://ieeexplore.ieee.org/abstract/document/5987118/authors#authors
   This paper provides a simple overview of Dijkstra algorithm as well as visualizations from simulation that may be used for inspiration with respect to visualizing our own test cases.

   (ii) "A Multiple Mobile Robotics Path Planning Algorithm Based on A-star and Dijkstra Algorithm" → https://gvpress.com/journals/IJSH/vol8_no3/7.pdf
   This paper includes a section that outlines the limitations of Dijkstra algorithm, which is always a good consideration to be made. It also compares Dijkstra to A-star path planning and presents an algorithm that is a combination of the two. This idea may be utilized when developing our own path planning algorithm.

(iii) “Improved Dijkstra Algorithm for Mobile Robot Path Planning and Obstacle Avoidance” → https://cdn.techscience.cn/ueditor/files/cmc/TSP_CMC-72-3/TSP_CMC_28165/TSP_CMC_28165.pdf
This paper presents a modified Dijkstra algorithm that includes avoidance of moving objects, which will likely serve as a foundation for our own development. A clear overview of the algorithm is given, as well as some experimental procedures that may help shape how we approach testing on the jetbot. The section concerning simulation and testing is also very coherent and visualizes the path planning solution very clearly.

5. The simplest solution would be to have both jetbots, prior to moving, determine their respective paths globally (a single time before movement begins). In this case the initial position of each jetbot would simply be included in each respective obstacle list, and the jetbots would be treated as static objects. This method is very inefficient because it staggers the movement of the jetbots, with one completing its path to the goal followed by the other. A better solution is to have both jetbots start at the same time and communicate their own updated positions to the other jetbot. This may be as simple as making a new ROS node that each jetbot publishes its current (and potentially future) position to, which is used by both jetbots to determine cases of collision. Of course considerations will have to be made for deadlock situations, so that the jetbots are not stuck perpetually trying to avoid eachother.

6. Second Report Goals:
   (i) *Safe goal*- Develop good and diverse test programs for Midterm 1
   (ii) *Moderate goal*- Complete Midterm 1
   (iii) *Ambitious goal*- Complete Midterm 1 and develop test programs for multiple jetbots/dynamic object avoidance.

   Final Report Goals:
   (iv) *Safe goal*- Complete Midterm 1, show adequate tests and simulations for multiple jetbots
   (v) *Moderate goal*- Implement a new object avoidance algorithm and demonstrate with two jetbots that are able to avoid eachother while finding their goals
   (vi) *Ambitious goal*- Improve dynamic object avoidance to include moving Aruco markers.

   The team members will work together on finalizing the Dijkstra algorithm and getting it to run on the jetbot but they will divide the test programs between the two of them. If the team is not successful in meeting their safe or moderate goals, they will provide a document outlining the reasons for which the goals were not met, as well as attempted solutions to these issues. The document will also outline a future plan for how the project would be completed were these issues not encountered.

7. Measurable/Quantifiable Goals

   Second Report:
   (i) Generate test cases for random object, goal, and initial jetbot placement on arbitrary grid size. Develop and run planning algorithm and visualize results for static object placement.

(ii) In addition to (i) show video evidence of a jetbot properly navigating to a goal from multiple goal/object/jetbot orientations.
(iii) In addition to (i) and (ii) generate test cases for path planning which include one other object navigating to a goal in the same graph. Visualize results.

Final Report:
(iv) Same goals as (iii), potentially add new test cases to include multiple moving objects as well as deadlock situations.
(v) Video evidence of two jetbots concurrently navigating to shared and different goals. Show success for multiple initial goal/object/jetbot positions.
(vi) Video evidence of two jetbots successfully navigating to goals amidst moving Aruco markers. Potentially keep a log of jetbot and obstacle position in real time and later visualize actual paths taken.