Mechanical Engineering Department
ME 598 Introduction to Robotics
Fall 2016

ME 598: Final Group Project
Navigation and Localization

For this project, student groups are required to use navigation and localization to locate landmarks, decipher a code, and recite the code. To complete this advanced task, the team will need to use sensor/image processing, odometery, and geometric relationships.

Test Space Layout

Figure 1 illustrates the testing arena. Landmarks include four orange cones, four cones that can be individually set as purple or yellow, a target cylinder, a black wall. The arena bed is comprised of 24 inch square black gym-floor tiles and white navigation lines.

The four purple/yellow cones will define a four bit binary code that must be determined by the robot. A purple cone designates a binary zero, while a yellow cone designates a binary one. The code will be randomly set *by the professor* during each trial. It is read from left to right, e.g. the leftmost purple/yellow cone is the most significant bit (MSB) and the rightmost cone is the least significant bit (LSB). The cones containing the binary code are hidden from the start position by a black wall.

The robot must autonomously navigate through *all 12* perimeter tiles as well as view and decipher the binary code. After completing this navigation task and deciphering the code (in any desired order), the robot must then navigate to *within six inches* of the target cylinder and report the four bit code using a series of beeps, followed by a spin routine to designate completion. The goal is to accomplish this entire task in the minimum amount of time possible. A successful trial requires completing these objectives without any collisions or driving off the testing arena.

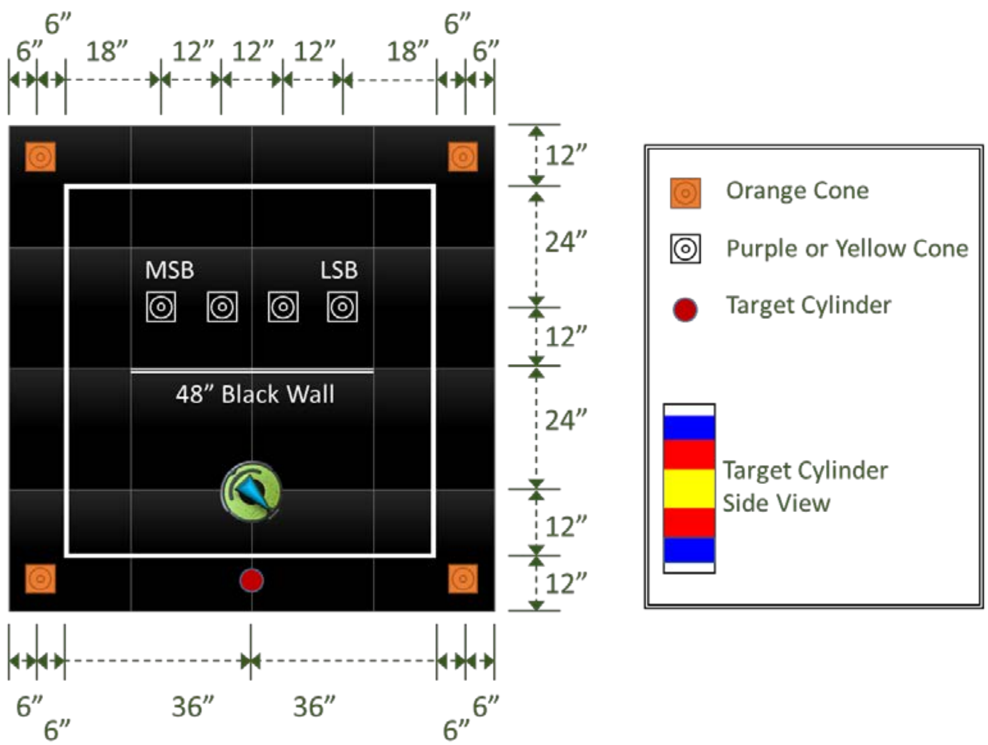

Figure 1: Overhead view of test space layout.

Functional Objectives

Trials will be run in a standardized fashion. Before a trial begins, the robot should be placed in the designated starting position in some random angular orientation. Once the trial initiates (the program is started), the robot will **beep once** and wait 10 seconds before carrying out any other operations. This is when the professor will randomly assign the four bit binary code by placing the appropriate cones in their designated locations. After these 10 seconds pass, the robot will **beep twice** and begin its routine to navigate around the perimeter tiles as well as to determine the binary code. These tasks will require navigation, localization, and obstacle avoidance. Once the navigation and code deciphering tasks are completed, the robot must navigate toward to within six inches of the target cylinder and report the code in a series of beeps, as specified. After reporting the code, the robot should **spin in place three times** to designate completion of the task.

The two teams with the shortest (fully functional) trial times will be awarded extra credit!

Additional rules:

- The robot must be programmed using MATLAB and must not use any third-party toolboxes or software not already utilized within the course.
- During operation trials, the robot must operate autonomously with no form of input from group members or other users.
- The test space may not be altered for any trial (e.g. adding ground markings or further landmarks).
- Binary codes must be determined using a visual processing algorithm, not by random guessing or similar approaches.
- The robot must not collide with any of the landmarks/walls/etc. at any point during a trial.
- The robot may not drive off the designated course area.

Specific Tasks to Complete

To accomplish the functional objectives, the following specific tasks must be completed:

- Create algorithms for recognizing and determining the difference between orange cones, purple cones, yellow cones, and the target cylinder.
- Create an algorithm that uses known geometry and the locations of the landmarks to correctly determine the binary code.
- Create an algorithm to navigate around the perimeter arena tiles and determine when all perimeter tiles have been traversed.
- Begin at the starting position and navigate through the test area while avoiding obstacles.
- Navigate back to the target cylinder while avoiding obstacles.
- Create a MATLAB function `binarybeep.m` that allows the robot to audibly playback the four bit binary number designated by the cones. See Appendix for specifications.

Extra Credit

The two teams with the shortest (fully functional) trial times will be awarded extra credit. Extra credit may only be earned if the project is presented on time and all other required aspects of operation are deemed complete.

Record and Upload a Video

Record a video of your team's best trial run. **Note that this DOES NOT replace the live demonstration that must be performed for the professor!**

- Upload the video to the class YouTube site, and e-mail the link to the professor (CC TA). Each upload **MUST** conform to these rules:
  - Title: ME 598 Fall 2016, Final Project, Group [assigned group number]
  - Description: Final Project – Navigation and Localization, Group [assigned group number]: [team member list (last-name alphabetical order)]

Username: StevensME598@yahoo.com
Password: intro2robo

Example ME 598 Fall 2016, Final Project, Group B7
Final Project – Navigation and Localization, Group B7: John Doe, Bob Smith.

Schedule a Live Demonstration with the Professor

All teams ***MUST*** schedule a live demonstration of the functional robot with the professor. The assignment is not complete until the professor approves live operation. A demonstration may be scheduled once the team has uploaded a best trail video to the course website.

Report Submission

A final report must be submitted via Canvas along with associated code. **The submission will be due within 24 hours of your live trial demonstration.** The submission must include:

- A complete detailed report.
  - How accurate/repeatable was your system?
  - **Discussion of live trials performed with professor.**
    - Trial times, level of performance, # successful/failed attempts, etc.
- A detailed explanation of your algorithms and methods.
- All `m-files` and code utilized (must be well-commented!).
- A link to your YouTube best trial video.
- A student feedback summary.
  - Explain any difficulties you encountered when completing the project.
  - What did you think of the project? How long did it take you to complete (in hours)?
  - Would you recommend this same project for the future?

Appendix

The MATLAB function `binarybeep.m` must be created by student groups to accept the four bit binary number designated by the purple/yellow cones and play the associated code as a series of sounds. The sounds should be played starting from the MSB and ending with the LSB.

A binary value of zero, designated by a purple cone, should be played as a 220 Hz beep in the key of A. A binary value of 1, designated by a yellow cone, should be played as a 329.6 Hz beep in the key of E. For example, a binary code of 0101 would be played as A-E-A-E.

Calling the function should be done like so: `binarybeep (serPort, binaryvect),` where `serPort` is the serial port connected to the robot and `binaryvect` is a four element vector containing the binary code in order from MSB to LSB.

For example, if the serial port is COM3 and the binary code is 0101, the function would be called like so:

```
binarybeep (3, [0 1 0 1])
```

Hint: consult the `iRobot Roomba Serial Command Interface Specification` and `RoombaInit.m` for ideas.