Autonomous Maze Solving Robot
Lawrence Lenoci, Joseph Bruno, Johanna Gordillo
{lml237, jdb318, johanna.gordillo}@scarletmail.rutgers.edu
Advisor: Assistant Professor John McGarvey

Abstract
This project aims to explore the difficulties that arise as an autonomous vehicle (AV) tries to traverse a road without the aid of off-board technologies such as cloud-based servers, 3D-mapping, or GPS. In essence, the vehicle is reliant only on locally gathered data and its onboard computational power.

Motivations and Objectives
Motivation:
There exists a lack of research and development into the technology that could potentially allow an autonomous vehicle to navigate unmapped and rural areas. Most current autonomous vehicle technology is fully reliant on GPS and preloaded 3D maps. But what happens when the vehicle is unable to communicate with those servers?

Objectives:
1. To create a small-scale robot that can solve a wall maze.
2. To build an efficient and safe system that will not crash into walls due to poor steering.
3. To have the robot solve any maze orientation.
4. (Given time) Re-solve a particular maze without turning down any dead ends.

Research Challenges
Selecting the optimal proximity sensors was a pivotal step in the design process. Common sensors used in AV designs are RADAR, LIDAR, infrared, and ultrasonic. Ultrasonic sensors were selected because of the material used for the maze and the smaller lower bound on the sensing range.

Understanding the control theory necessary for the robot to navigate the maze was an unforeseen aspect of this project. A PID controller was researched and implemented into the Arduino code.

Acknowledgement
We would like to thank Assistant Professor McGarvey who gave unparalleled advice and ideas for the completion of our project.

Methodology
- Robot: Must take into account weight distribution and number of wheels and sensors.
- Maze: Allow walls to be reconfigured on the base board into different orientations.
- Code: High level algorithms for decision-making, wall detection, and error correction.

- Solder sensors, motor shield, and motors to Arduino Mega.
- Fabricate chassis and mount all components.
- Construct maze base board and walls.

- Implement algorithm designs into Arduino IDE using C++.
- Test code on robot, fine-tuning constants and algorithms for speed and turning.

Results and Further Development
- Results:
The robot was successfully able to navigate multiple mazes with minimal collisions with the walls.

- Further Development:
Future plans for this project include the Advanced Learning Option detailed in No. 4 of the objectives. One possible approach to this include adding a rotary encoder to the wheels to record wheel rotations or adding a position sensor.

References