The Gauntlet

Sam Wisnoski and Esther Aduamah

Link to neato vid:
https://drive.google.com/file/d/1Dn6on7AbxUASv2Oy1qWFpPj7JmD4Ir11/view?t=6
Link to Matlab code in Google Drive:
https://drive.google.com/file/d/1gqWokT9sMUPQo4yjvGJDVi1K86OfWfmA/view

Methodology

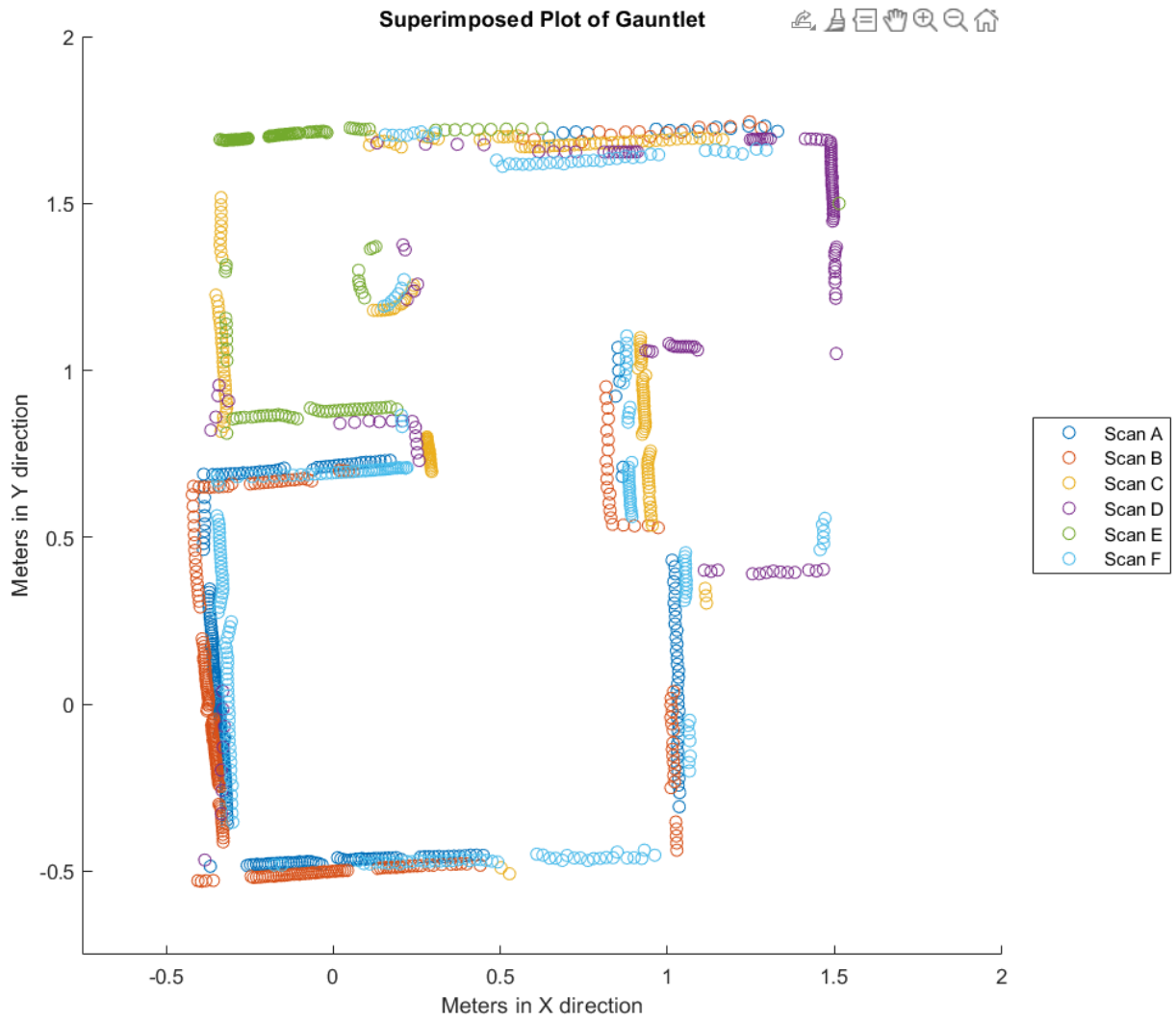
The goal of 'The Gauntlet' project is to go through a series of obstacles to tap the Ball of Benevolence (BoB) with the Neato. Creating a theoretical path using LIDAR scans and gradient descent methods to find the steepest descent path (this sink is the BoB) will accomplish this.

We took multiple LiDAR scans throughout to create a map of Gauntlet and then superimposed them onto each other. From these scans, we created points in polar coordinates then were then transferred into cartesian coordinates. After looking at this superimposed plot, we recognized where the BoB would be placed. Using the equation:

$$f(x,y) = ln\sqrt{(x-a)^2 + (y-b)^2} - C * ln\sqrt{(x-a)^2 + (y-b)^2} + \ldots$$

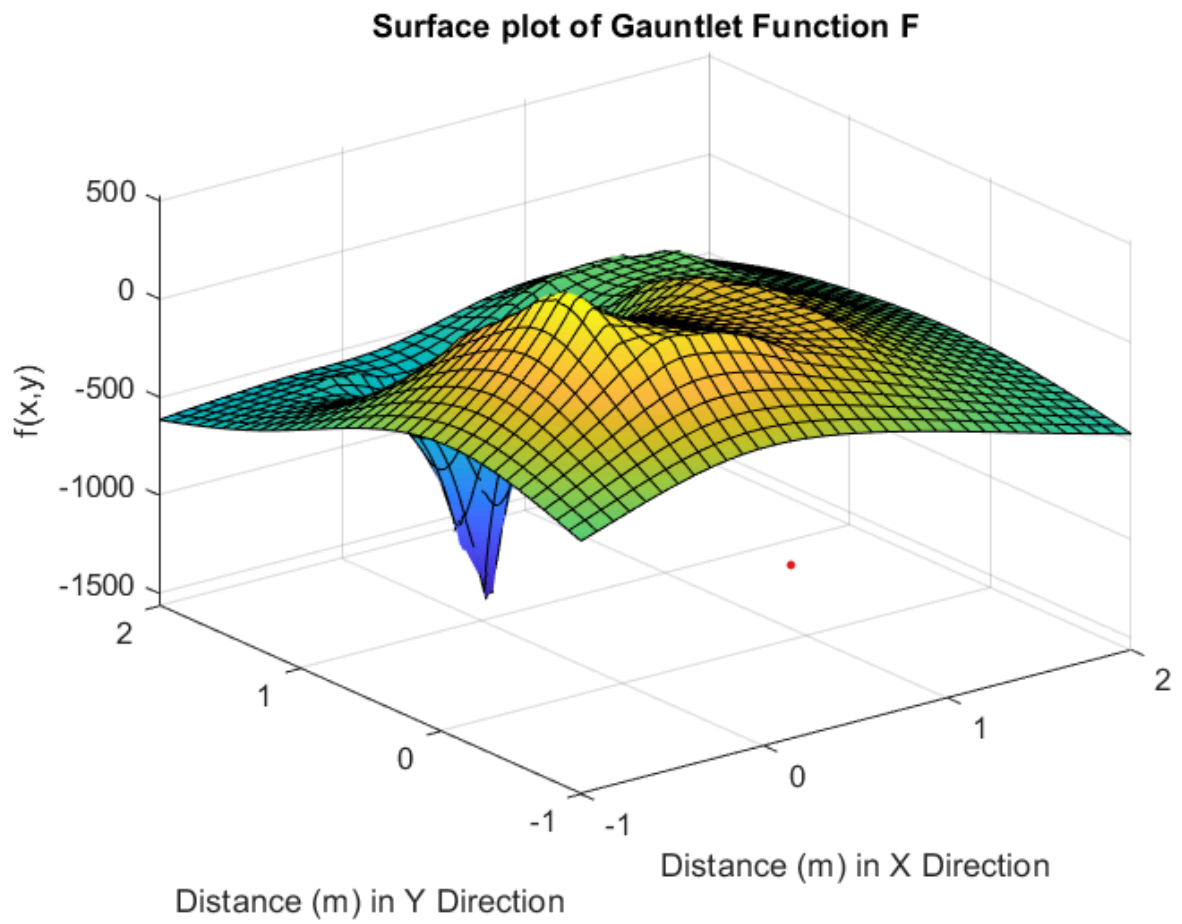
where a and b are the x and y cartesian coordinates of a point from our LiDAR scan, we set up our potential field equation. Iterating through each point gathered via our LiDAR scans, we classified each point as either a point on the wall or a point on the BoB. If a point was on the wall, we counted it as a source, or positive ($ln\sqrt{(x-a)^2 + (y-b)^2}$). Each point on the BoB was a sink, or negative ($-C * ln\sqrt{(x-a)^2 + (y-b)^2}$). Since we had far less sink points and needed the neato to navigate directly to the BoB, we multiplied each sink point by 25. This gave us our final function f(x,y).

Next, we created a contour map of f(x,y), and found the gradient of f(x,y). Finally, we were able to use a gradient descent algorithm by reusing our flatland code to navigate through our contour map. This step required tuning many variables: we ended on a lambda value of 0.00024 and a delta value of 0.99. After successfully tuning our values to this course, we were able to successfully run the code and complete the gauntlet.

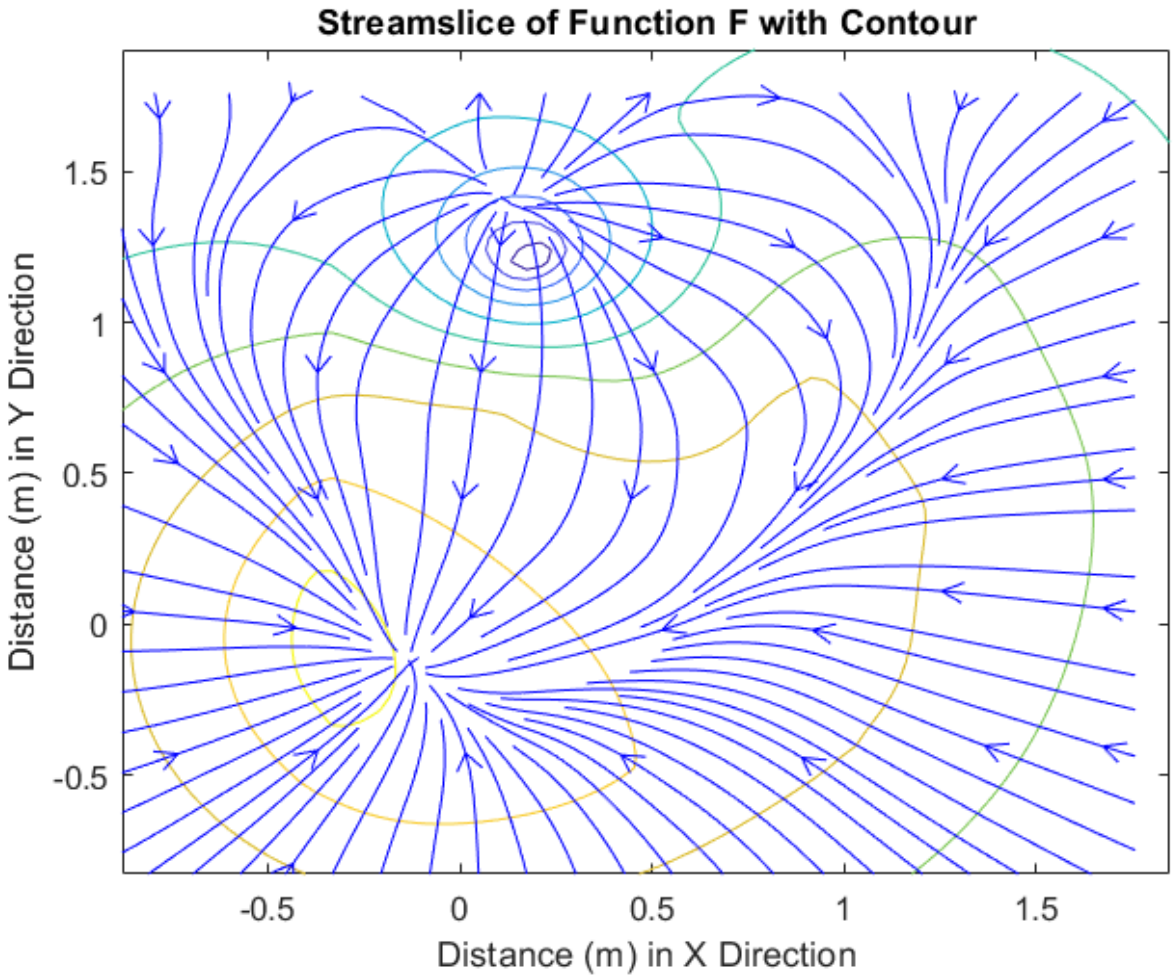


Above is a map of Gauntlet using LIDAR scan data. We took six scans at different locations and angles to get the full readout of the Gauntlet.

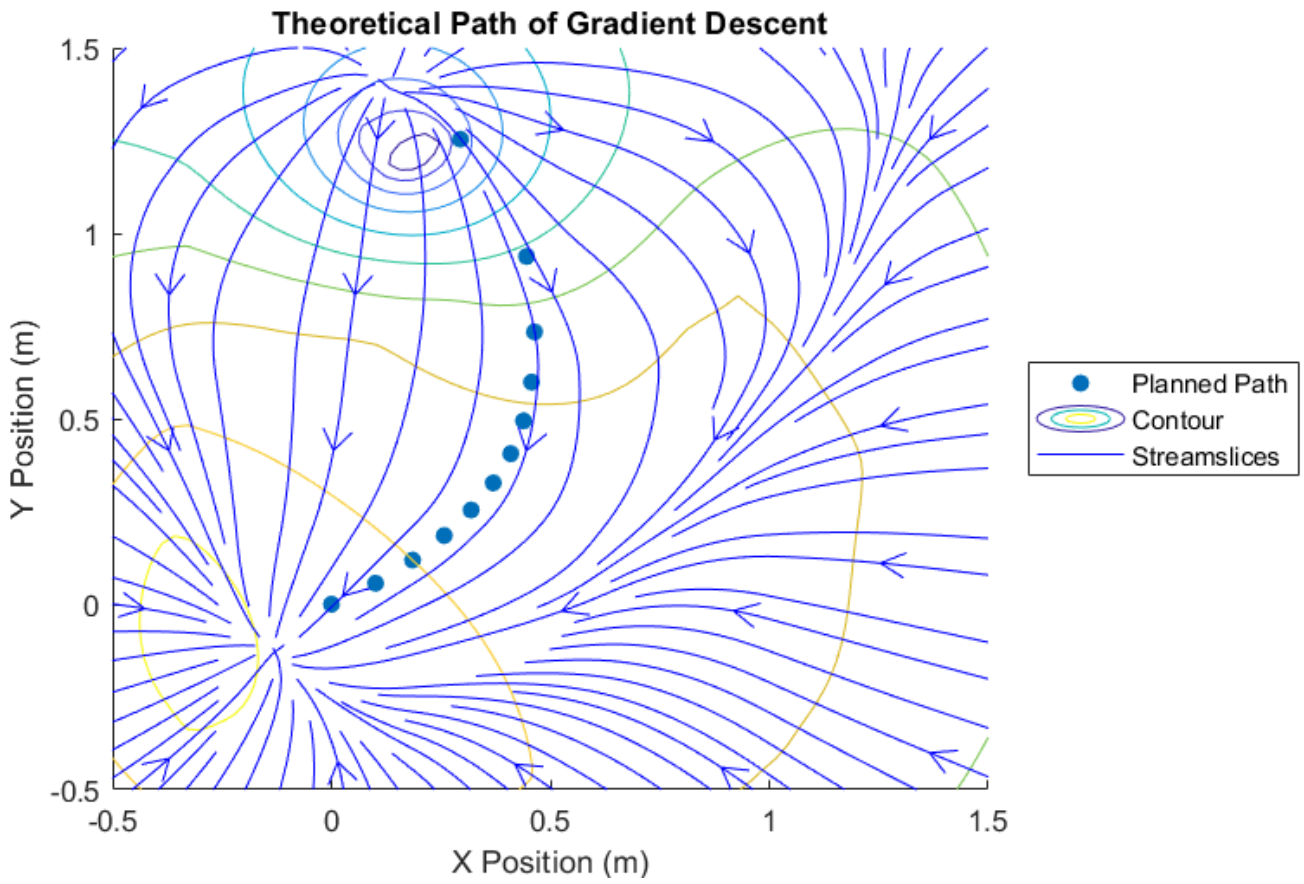
Position A: (0,0) 0 degrees
Position B: (0,0) 45 degrees
Position C: (0.5,1) 65 degrees
Position D: (1,1.5) 210 degrees
Position E: (-0.1,1.5) 0 degrees
Position F: (0.5,0.5) 0 degrees



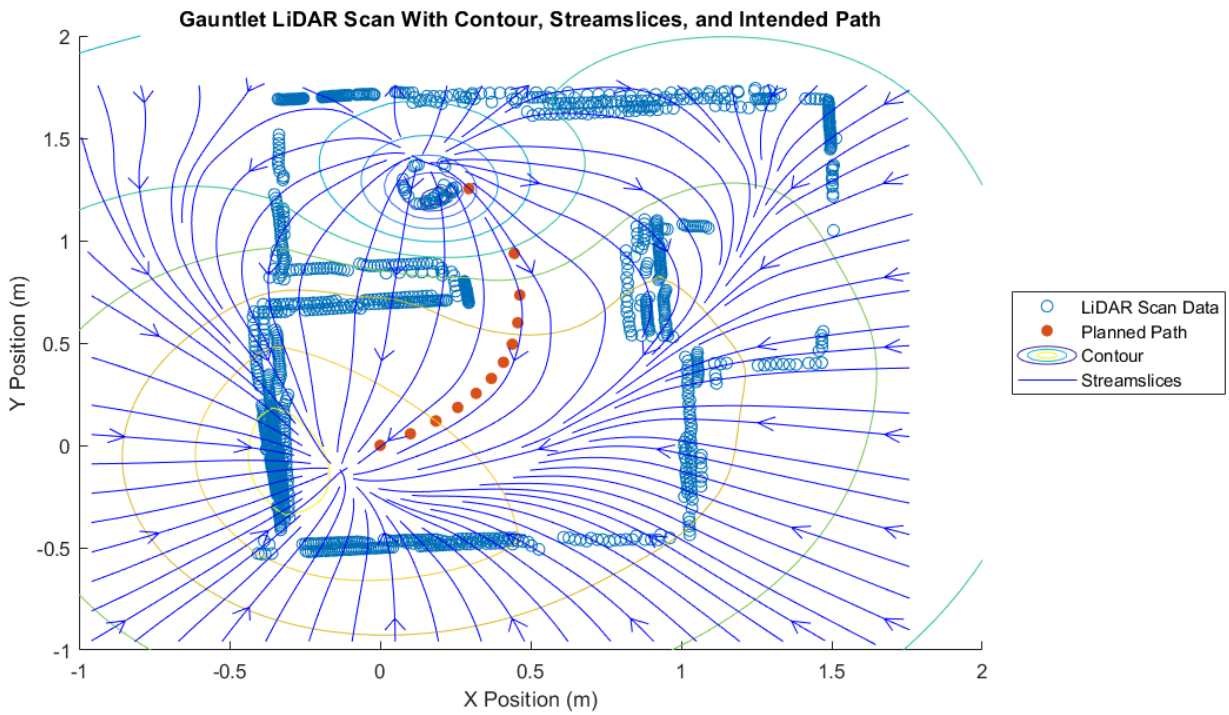
Above is a 3-D graphical representation of the potential field developed for navigating the gauntlet. Each point above zero f(x,y) is because of the point's proximity to the walls of the gauntlet, while the large sink corresponds to the position of the BoB.



Contour Plot of the Gauntlet based on function F. Function F is based on a list of sources and sinks - for each LiDAR scan, each wall point is a source and each point on the BOB is a sink. The streamslices are based on the gradient of F.



Above is the contour map potential field that was developed for navigating the Gauntlet; along with the planned path for the Neato to follow and the streamslice map.



Above is the map of the Gauntlet using LiDAR Scans overlaid on a contour map that was developed for navigating the Neato; along with the planned path for the Neato to follow and the stream slice map, further demonstrating how our Neato will go through the Gauntlet.

Our system worked well for the most part. We had to play around a lot with different variables - for example, the ratio of how strong the BoB sink should be in comparison to the strength of the sources. Our system also changed significantly based on the lambda value used for path planning. So while our system worked very well for this specific task, it would likely fail other similar courses until the necessary variables had been properly tuned.

Total Time to Run Path (based on video): 54 Seconds

Total Distance: (time moving linearly)*(linear velocity) = (17 seconds)*(0.1 meters/second)
= 1.7 meters