For the robot competition and moving on from lab 8, the potential fields application has been expanded to include as many possible on a cartesian type plane cases. Previously, in prior labs, the potential fields algorithm did not account for the robot in other quadrants of a Cartesian plane with respect to its target. For now, given a known location and a target and regardless of which quadrant, the robot should be able to adjust accordingly, its heading. These improvements or edge cases were made in the python code within the potential fields section, beginning on line 340.

The potential fields functions produce a repulsive angle which is either added to or subtracted from the robot’s current pose depending on which set of sensors is active. The result of this repulsive angle translates to or utilized in the realignment function. If the front and right sensor is active, due to predefined thresholds, then a course correction is made away from the obstacle, counter clock wise. The robot then traverses away from the obstacle based on the magnitude of the repulsive vector. If the front and left sensor were active then similar logic would occur with the exception of a course correction clock wise to the robot’s orientation.

In addition to the boarder cases made for potential fields, a map was implemented to help verify and track the global position of the robot. Unfortunately, the map is not an exact representation. The robot will use range sensors to update walls and objects as it traverses the map. To add, we have also added the use of a LIDAR to add to its sensing capabilities.

Communication protocol, the tiva board is automatically sending the raspberry an 8-byte packet every tenth of a second. The pi sends motor control commands under specific conditions such as range sensor distance is below a certain threshold. Turn commands are sent based on the potential fields functions and realignment function.