Dale Myers 0942590

MILESTONE 4 REPORT

This week I proposed and helped implement the new control code for the robot. Most of the work was making the new code integrate well with the current system.

Starting from the last milestone, the first thing I worked on was designing the new robot control interface. The old one had become clunky, difficult to use, difficult to understand and difficult to test. We needed a new interface, one which we could use simply and would be easy to use and understand.

We can send one integer over bluetooth at a time. We needed to make the most of this in order to keep our robot as accurate as possible. I designed a "packet" which seemed optimal. This consisted of one byte being used for kicker information (currently only whether we should kick or not), one byte for our control code (should we move forwards, backwards, arc, stop, rotate, etc.) and the final two bytes were used for the magnitude of the control code (how much should we move/rotate by). We discussed using a single byte for magnitude but due to problems with negative numbers and the desire to keep the implementation as simple as possible we would only be limited to 127 values. We felt that this was too small. Using two bytes let us have a range of zero to 32768, a much more acceptable figure.

The code for the robot was rewritten to use the new code (removing any bitshifts that we had previously and using array indices instead). We did find some issues with the kicker and the bump sensors. When we sent a command to kick it did it over and over again until it received a command to stop, and when the bump sensors were hit everything else locked up and the robot would not respond. Wiktor and Tom solved those issues. The final thing to change was strategy. This control method worked on the basis that the robot worked on a single instruction and so the instruction had to be as relevant as possible. This meant removing any sleep statements from strategy so that it generated plans as fast as possible. Upon testing, the new control interface was a huge success. The robot did exactly what the computer told it to do, something which it couldn't before.

Other work done for this milestone includes updating the plan monitor overlay to use colours to distinguish between different parts and objects in the generated plan as well as working with the new A*, and writing part of the milestone 4 code.

Changing the overlay was relatively simple requiring minimal effort. Most time was spent making sure that it worked with the slightly different representation of nodes. The milestone 4 code on the other hand was slightly trickier. Others in the team had tried to implement this and failed. I simply used the basic assumption that if the ball was not moving, when it did it would do so in the direction that the kicker was facing. I also made sure that the commands were non-blocking so that if the ball changed speed or direction (due to the pitch) it would be able to cope.

The rest of my time was spent working on minor changes such as bug fixes.