Matthew Rodin (260623844)
Tiffany Wang (260684152)
Group 22
January 28th, 2016

LAB 1: WALL FOLLOWER

Data Analysis

- The bang-bang controller did not keep the robot at a constant distance bandCentre from the wall; instead, it made sure that the robot stays approximately bandCentre away in an oscillating fashion. In fact, the robot is programmed to only adjust the speed of its wheels at every ultrasonic reading in order to get closer to the band, yet this correction of speed is always the same regardless of the error distance. As long as the ultrasonic does not read the data at the very instant it is at the band, it would just cross it and continue forward. Therefore, it is hard to be at a constant distance from the wall, and the robot would just oscillate from one side of the band to the other.
- The P-type controller, known as the proportional-type controller, holds the robots extremely close to the band. Similar to the bang-bang controller, the p-type controller changes the speed of the wheels in order to make the machine turn closer to the band. However, this adjustment is proportional to the distance from the band. Hence, the turning angle decreased as the robot reaches the center. It then smoothly follows the wall at the desired distance.

Observations and Conclusion

1. **Errors the ultrasonic experienced?**

- Our biggest error to overcome was the included *gaps* in the course that would cause our robot to alter its path.
- It is important to note that the ultrasonic sensor detects a max reading of “255” when passing by a gap AND while turning around after completing one side of the course. Therefore we needed to find a method in which the robot would continue straight at a gap but rotate at a turn.
- Our first idea was to implement a counter, that would be initiated (and stopped once a pre-determined threshold is reached) when the ultrasonic sensor detected a reading of “255”. The goal was that this wait time to increment the counter would be just enough for our robot to pass the gaps but still make turns when it needs to. Our second idea, given the fact that there were no speed requirements, was to decrease our robots speed to an almost crawling rate. Whenever the ultrasonic sensor detected a reading of “255” the robot turned slightly. We chose to implement the ladder.
- A secondary error that we encountered was, during p-type, when the robot needed to perform sharp concave turns (i.e. too close to the wall and/or corners). The robot would still accelerate and result in hitting a wall. We concluded that this was a direct result of the sensors slow reading nature. Therefore we fixed this issue by adjusting the speeds of the motors in order to create sharper turns.

2. **Does the ultrasonic sensor produce false positives, false negatives or both?**

- If the robot is too close to the wall, the sensor would give a reading of "255" and not "0".
- We found that as we moved the robot further and further away from the wall, the sensor would often "255" instead of the actual distance. We decided to implement a threshold of 75 (because after that distance it would have trouble detecting false negatives)

Further Improvements

- **Improvement 1 - Software:** When implementing the P-type controller, we inherently adjusted the motor speeds by calculating an "adjusted speed". However, we never reset the speed of the motors after adjusting it. Therefore, we saw sometimes the robot turn sharper than others. Eventually, the speed would reset when we enter a new if statement condition of the code however we could have improved our software component by implement a way to reset the motor speeds if the range of values remains the same.
- **Improvement 2: Hardware:** Using infrared sensors rather than the ultrasonic sensors which came in the Lego kit. IR sensors would have a faster reaction time (thus improving the robots trajectory), a lower cost and a narrow beam width.
- **Improvement 3: Hardware:** Both the Bang-Bang and P-type controllers would have been greatly improved by adding a *second* ultrasonic detector. This would no doubt increase the robot's ability to retrieve a more accurate value of its distance from the wall.
- **Other type of controller?** A PID controller would have ben even more affective than the p-type of bang-bang controllers implemented during the lab. A PID controller can adjust speeds by constantly calculating/manipulating an error value. The goal is to minimize this error over time.