

# Lab 1 - Wall Follower

---

John Wu - 260612056

Alex Lam - 260746239

January 17, 2017

## 1 Objective

The objective of the lab was to create a vehicle that could navigate around a sequence of cinderblocks making up a wall containing gaps, concave corners, and convex corners, without touching or deviating too far from it.

## 2 Method

Template code for wall following was modified to:

- Avoid getting confused by gaps.
- Turn concave corners.
- Turn convex corners sharper.
- Implement both P and Bang-Bang Controllers for the above requirements.

### 3 Data Analysis

- **Did the bang-bang controller keep the robot at a distance Band Centre from the wall?**

No it didn't keep the robot exactly at a distance BandCentre from the wall at all times. Since the sensor only returned data a fixed amount of times a minute, if it failed to return data at the exact moment when the robot was at a distance BandCenter away from the wall it would continue to correct the course of the robot. This meant it was very hard to stay on the bandCentre with the bang-bang controller. In addition, we also added a rotating sensor feature in order for our robot to be able to know how far it was from both the side and front walls. This allowed us to have smoother turns, but at a cost of higher oscillations. This was due to the sensor readings from multiple angles in its rotation at parts when the road was simply straight.

- **Why is it expected that the robot will repeatedly oscillate from one side of the band to the other with the bang-bang and p-type controllers?**

Both the bang-bang controller and the p-controller was programmed to stay approximately BandCentre distance away from the wall in an oscillating fashion. This oscillating path was expected and was caused by the speed of our wheels only being adjusted at every ultrasonic sensor reading. The slight delay in responding to each sensor reading caused the robot to deviate slightly from the course. In addition, as mentioned above, if the sensor failed to return data at the exact instant the robot was on the band, the robot would continue to correct its course and cross the band before realizing that it was not on the band at which point the process would repeat. This slight imperfection of the sensor is what is expected to cause the oscillating motion of the robot.

### 4 Observation & Conclusion

- **What errors did the ultrasonic sensor experience?**

The ultrasonic sensor experienced problems when it was significantly close to the wall (5cm or less) or if it was significantly far from the wall (255cm or more). In both cases, the distance results in an infinite reading.

- **Were these errors filterable?**

We used the rudimentary filter method provided by the code template in order to filter out these types of values. The result of this would consist of the vehicle continuing to move in its regular path. This movement was valid for when the sensor was very far from the wall, but not for when it was very close.

- **Does the ultrasonic sensor produce false positives (i.e. the detection of non-existent object), false negatives (i.e. the failure to detect objects), or both**

As mentioned above the sensor would generate a false negative when the wall was much too close to the wall (<5cm in general). However, we did not experience many problems with the ultrasonic sensor in general.

## 5 Further Improvements

- **What improvements could you make to both the physical or software designs to improve performance of the wall follower? (At least 3 are needed)**

**Physical Improvement:** One of the observations with our vehicle was that it needed to make very wide turns. By making the width of the vehicle smaller, we could have reduced the necessary radius in order to make a turn.

**Physical Improvement:** The robot is currently front heavy as the extra weight of the motor and sensor tips the vehicle. Balancing the weight of our vehicle would have allowed it to be more stable. Sliding the motor and sensor towards the rear of the vehicle is a relatively simple fix that would have improved our robot.

**Physical Improvement:** We had to make additional adjustments to our error value, because our sensor was placed near the middle of our vehicle. By moving the sensor to the edge of the vehicle, we could have solved this problem.

- **Are there any other controller types that may have better outcomes than the bang-bang and p-type?**

An integral controller takes the error values and integrates them over a period of time. By taking an average error over longer period, our vehicle would have a smoother error correction process. As a result, using an integral controller would result in less oscillation of our vehicle.