# Project 2 Report: Adaptive Cruise Control and Autonomous Lane Keeping

## 1. Adaptive Cruise Control

### 1.1 Problem Statement

In this problem, the goal is to keep the vehicle 30 cm's away from an object ahead. The primary objectives are to stop at the correct distance with low cyclic vibration about the stopping point. Additionally, the vehicle must run down the ramp as quickly as possible while remaining under control.

### 1.2 Technical Approach

In order to address this problem a state space model of the system is first constructed. We will use an ultrasonic sensor to measure the distance from the vehicle to an ahead object. The system has both control input and measurement. The output from our controller to the vehicle is not acceleration or velocity but is throttle. For the sake of our controller, we treat the throttle like an acceleration control and model the system as the following.

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \qquad \textit{Eqn. 1}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) + u(t) \qquad \textit{Eqn.2}$$

This system model indicates that we are measuring out relative position and each state of our vehicle is a function of the previous states velocity and its acceleration. In order to control the system, we use a control model of:

$$u(t) = -Kx(t) \qquad \textit{Eqn 3.}$$

$$x(t) = (x_d - x_{sensor}(t)) \qquad \textit{Eqn 4.}$$

In order to determine the value of K we first use a Routh array to determine the values of K which will maintain system stability. Our Routh array is:

$$\begin{bmatrix} 1 & K1 \\ K2 & 0 \\ K1 & 0 \end{bmatrix}$$

Indicating that K2 and K1 both must be positive for the system to be stable. In order to evaluate what to use as a K value, we utilize a linear quadratic regulator which attempts to balance a cost function based on minimizing control input and maximizing accuracy. Our LQR constants, and Q and R are modified according to our desire for balancing. The K value determined by the LQM is then used for our controller.

### 1.3 Hardware and Software Implementation

This system was implemented through a simple single sensor configuration. The sensor did not utilize any filtering due to the complexity of the state space model and unknown output control. In practice our values were not significantly unstable to warrant the need for this type of control. A significant portion of the difficulty in controlling the system was caused by the low resolution of the control of the ESC. Through testing we identified that the ESC was not responsive to control inputs less the +/- 5 and therefore we needed to give the ESC a minimum control input outside of this range. Additionally, at the minimum control input the ESC caused the vehicle to move at a high speed. This meant that we needed to implement a secondary control strategy in order to slow the vehicle down to a more controllable level. We decided to perform this task with a pulse width modulated controller. This controller operated by interpreting the control input from our model as a percentage on time for the vehicle motor. This percentage on time allowed us to quickly turn on and off the motor. The result of turning the motor on and off quickly was a significantly slowed and more controllable velocity. Additionally, the PMW allowed us to precisely modulate our control input in order to make our vehicle get closer to the task distance without having to reverse. The vehicle was fully stopped when reaching the desired distance without any additional control thanks to another difficult-to-address property of the ESC response. The ESC required not only a minimum control input but also must receive that input value for a minimum amount of time. This allowed our vehicle to stop without additional control because our model output values for on time less than the activation threshold at distances very close to the desired distance.

Through tuning this control we were able two also increase the responsiveness to reverse control. This was necessary in order to counteract the effects of gravity and traction that caused the vehicle to be more likely to slip when reversing. This increase was needed to allow the vehicle to back up at the start of the course.

The velocity measurement required for the model-based control proved to be relatively difficult to obtain stably. In early attempts, the time steps were highly stepped, likely due to the digital nature of the output of the microseconds command and inconsistencies in unfiltered data from the sonar sensor led to wide variations. Eventually, through experimentation we were able to resolve the issue by filtering the velocity to 0 at low changes in distance. This lessened the input to the motor and led to much smoother operation especially at very low speeds.

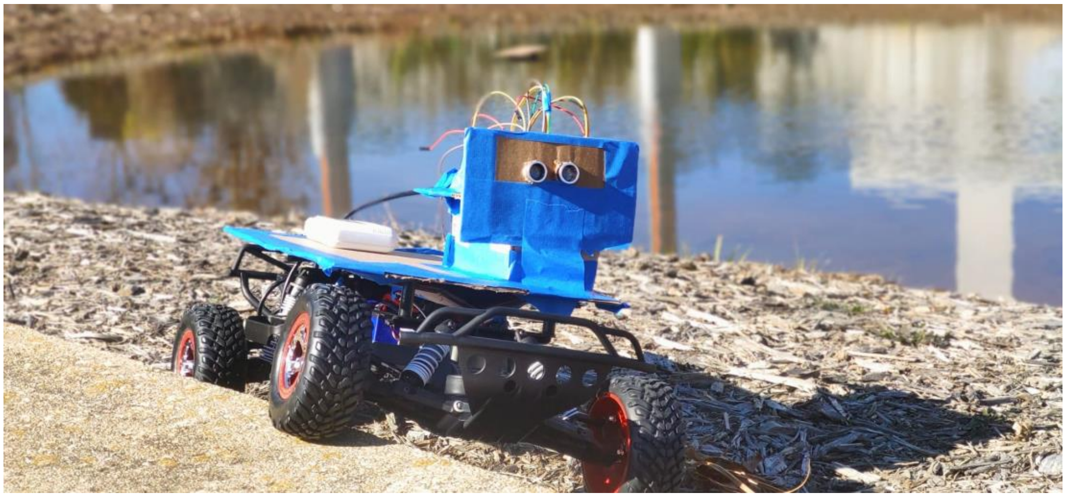

Figure 1. The assembled vehicle. The longitudinal sensor is clearly visible from this angle.

## 1.4 Experimental Results

While moving the robot we were unable to collect data from the sensors due to the length of the cord. This section will contain some data produced while the car itself was turned off. During our running trials the system was able to backup when the obstacle was brought closer, though it is important to note that if the object was brought towards it with high velocity the vehicle would reverse away too quickly. This problem could be addressed with tuning, however it was not necessary due to the context of the task. An additional issue faced while tuning our longitudinal control was the significant changes in power output based on the vehicle battery level. We initially tuned our longitudinal control with a battery charged to less than 50%, and after that charging the battery the vehicle was much faster than expected.

Figure 2 displays a sample of data measured in an artificial scenario where the vehicle is powered off and is manually moved close to and away from the wall at a variety of different speeds. This chart allows the control output to the ESC to be visualized, displaying how the vehicle attempts to move in order to center on the desired input. This model relies on the modeled output of the system, however, meaning that the output distance combinations reflected in this chart are not necessarily the same as those that are experienced while the vehicle is fully under its own control. The model function is only a function of the velocity and distance of the vehicle relative to the object in front of it. These values are not affected by simulating artificially, however, the LQR formulation relies on an expected model state related to the acceleration of the vehicle in time.

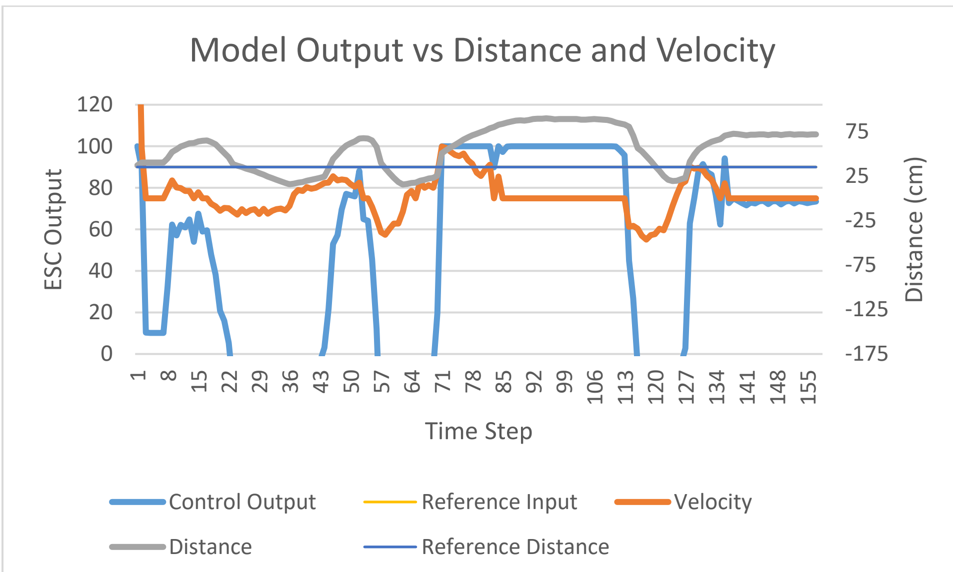


Figure 2. Model output vs longitudinal velocity and distance

The model output was passed through the PWM filter in order to output an actual command to the ESC without raising the vehicle velocity too much. As seen in figure 3, the PWM is highly sensitive to small variations in the output of the model, allowing it to control speed very precisely. The figure also demonstrates the requirement for the PWM off time to be much higher than the PWM on time in order to slow the vehicle sufficiently. Finally, the chart also demonstrates the different scaling of the PWM off and on time relative to the direction of travel, a feature necessary to allow the vehicle to utilize more accelerative input in reverse.

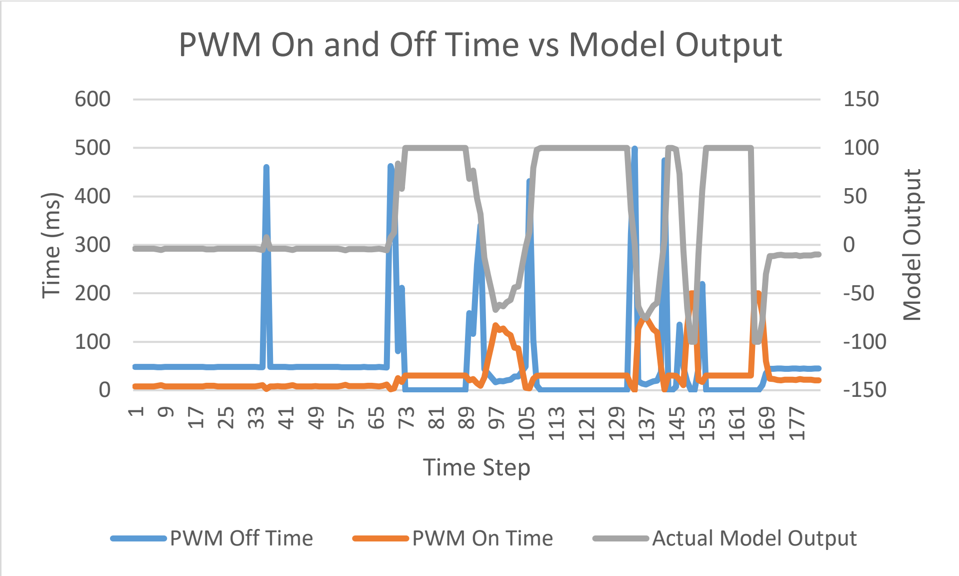


Figure 3. PWM on and off time vs model output.

## 2. Autonomous Lane Keeping

### 2.1 Problem Statement

In this problem the primary goal was to keep the vehicle within a specified lane. The lane was defined as a specified distance between the walls of the course. A secondary goal was to have the steering settled while the vehicle is stopped.

### 2.2 Technical Approach

In order to address this problem, we utilized a single side mounted sonar sensor to constantly take distance readings from the wall. These distance readings for enter to ask control inputs into a PID controller. The PID controller utilizes a standard, non-model-based equation with 3 tuning parameters in order to output a control signal. The PID controller was necessary in this case because the steering geometry coupled with the factors of vehicle weight distribution, variable ground friction, and complex geometry. There are many different approaches to tune a PID controller. We began by utilizing the Ziegler Nichols method. the PID controller attempts to maintain the desired distance from the wall by pointing the steering in the opposite direction. It is capable of proportionately varying the steering angle in accordance to the degree of steering output needed. While writing our software, we decided to use an existing PID library rather than directly using the PID modeling equations.

During our initial testing we had significant issues with the sensor seemingly losing its input signal. We isolated the problem to a short between the power and signal wire. After turning the connectors such that they were well isolated to each other we continued to have this issue. We removed the sensors negative ground wire and the sensor input data became more reliable. Further investigation is required to identify why the sensor continued to function as expected when it was no longer connected to ground.

We purposefully decided to not attempt to identify lane center from more than one sensor. More than one sensor could have been used in parallel to get a more accurate measurement but would have done so at the expense of execution time. We found execution time to be extremely important in smoothening our input to the ESC and therefore were intentional about reducing excess operations wherever possible.

## 2.3 Hardware and Software Implementation

In order to implement the steering control, we mounted a single sonar sensor on the left side of the vehicle. This sensor was read immediately after the front distance sensor. Through tuning KP we were able to achieve acceptable performance and maintaining the lane however the system exhibited too much steering input causing the vehicle to travel wildly back and forth between the lines. We address this issue by reducing the available steering input to a narrower scope. Additionally, we increased the value of the KD parameter in order to increase the settling time. We believe that this allowed the model to give less steering input when the vehicle was close to the center of the lane.

## 2.4 Experimental Results

Like in the case of the longitudinal control, we were unable to take measurements from the steering while the vehicle was in motion. We tuned our steering as necessary by sending the vehicle down the track changing the parameter slightly and then sending it again. Our steering control was able to maintain the vehicle within the specified area during the final test. The steering input was still too high while the vehicle was close to the center of the lane causing the vehicle to move back and forth rather than maintaining a consistent path. improvements could be made to the tuning of the PID model.

In order the display the control response of the steering control, an artificial scenario was executed by moving an object back and forth in front of the sensor. The control output and measured distances are displayed in figure 4 for reference. As visible in the chart, the steering output is overly responsive, giving maximum input as soon as the distance varies from the set range. This data agrees with the behavior seen when the vehicle is running. In order to tune this further, it is possible to tune the PID controller based solely on this output graph. The parameters could be tuned to lessen the control input as the distance reported by the sensor is close to the set distance.

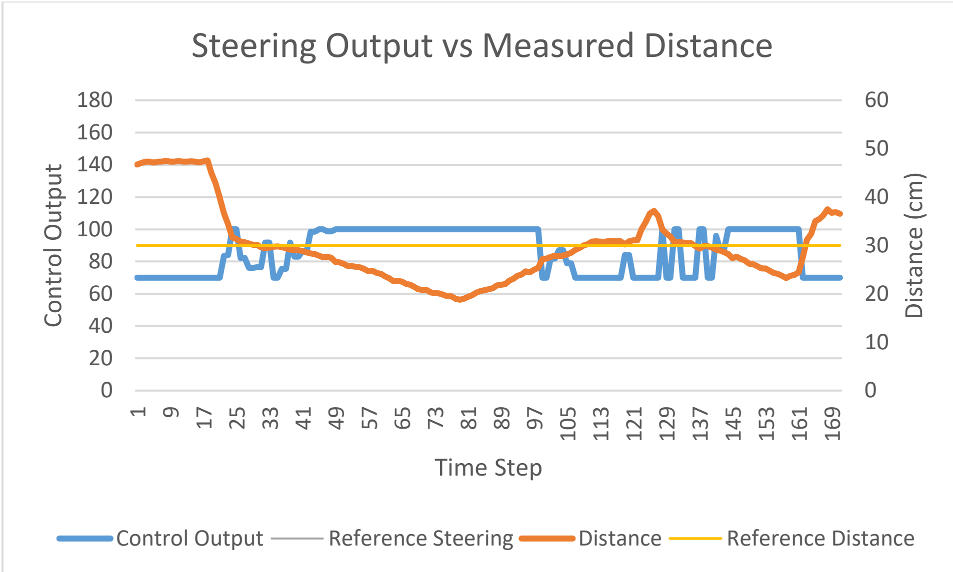


Figure 4. Steering output vs measured distance from wall.

## 3. Conclusions and Discussions

3.1 Conclusions (a summary the results of different approaches)

To conclude, our approach was very successful at completing the designated tasks in accordance to the assignment. Our longitudinal control was able to not only stop when it got to an obstacle but slow as it approached and stop before it passed over the line, allowing it not to have to pack up. Our very purposeful tuning of the PWM controller also allowed it to travel very smoothly and maintain good control over its speed. We were intentional about also decreasing our PWM stop time as much as possible in order to allow the control cycle to complete very quickly.

In our initial approach we attempted to use a Kalman filter to filter the values of the longitudinal sensor as the vehicle was moving. We attempted to do so by running trials with the vehicle in order to understand its accelerative output relative to the control input. We did this by taking to longitudinal distance measurements, deriving relative to time to get velocity measurements, and then deriving again to get accelerative measurements. We quickly ran into problems with this method however because our accelerations were extremely inconsistent and not related to the control input we were giving the ESC. We gave up on this attempt after this realization.

Since the sensors have a fairly limited accurate range, we also set our vehicle to travel at a set speed when the front sensor took too long to receive an impulse. We did not implement a PMW on this input so we were slightly surprised that the vehicle traveled at less than full speed without an obstacle within range. We suspect that the longitudinal sonar sensor was constantly detecting a distance from the inside of the wall on the outer radius, causing this behavior. We would like to have had the ability to experiment in an open environment in order to check why this was happening.

Additionally, through further exploration we identified that the acceleration also changed vastly with battery charge level.

Our steering control was sufficient to complete the task but could have been improved to be smoother.

3.2 Discussions (a comparison of different approaches, and potential future work to further improve each approach)

This project yielded thought into a variety of interesting other possible control techniques and improvements to the current design. Implementation of an IMU and order to measure acceleration would have allowed a Kalman filter to be implemented for distance measurements and would have allowed smoother input into the control function.

Additionally, the vehicle was not set up to account for any variation in environment from that in the test. If the vehicle reached the end of the wall and no longer had a surface from which to take measurements, it would simply turn continuously left. This could be avoided by utilizing two sensors and taking the difference in values between them instead of using just one sensor.

As shown in Dr. Jia's class, the use of multiple front sensors can also allow a vehicle to track objects based on relative position to each of the sensors. This would allow the vehicle to base its lateral control on the object in front rather than a wall to the side.

This project was a very valuable tool in learning about the difficulty of implementing controls in the real world. The rather straight forward state space and control models were minimal work compared to the difficulty of learning about how are system behaves and how we can control it without a direct output.

**Notice: The data in an individual report should be recorded individually. Students in one group can use the same hardware and software but cannot use the same data in the report.**