AIRBAGS - Autonomous Inertial Regulator Banishing Aimless Gaps of Space

Team Auto-bots: Daniel Little (3891), Hugo Son (3891), Ryan Taylor (3891)

Introduction and Problem Statement

Traffic congestion is an issue that impacts millions of people everyday. In 2021, the average US motorist spent 51 hours in traffic (Inrix 2022). Furthermore, Levy et al. (2010) found that air pollution from traffic congestion in the 83 largest American cities contributed to 2200 every year and cost the healthcare system $18 billion. Given this, reduction of congestion can be seen as imperative towards improving the health and environment of urban citizens. One manner in which congestion can be improved is to increase the *density* of traffic. The density of traffic refers to the number of vehicles per unit distance on a roadway. The flow rate of traffic, defined as the rate at which vehicles pass by a particular point on the roadway, is equivalent to the density of traffic times the speed of traffic. By increasing the flow of traffic, we increase the number of vehicles the roadway can transport, and therefore reduce zones of congestion where cars are not moving.

To increase traffic flow and limit congestion, we will outfit cars with a custom adaptive cruise control system called AIRBAGS (Autonomous Inertial Regulator Banishing Aimless Gaps of Space). At a high-level, our controller will direct a target vehicle to fill in large gaps that exist between it and its leading vehicle. Our controller will take input from the car's on-board sensors and translate that into the target vehicle's speed, with the goal of reducing the target vehicle's space gap.

Assumptions

There are several assumptions that need to be made for our controller. One of them is that preset parameters include the max lead distance, speed limit, following distance. Depending on these parameters, the controller will decide the acceleration value. The testing environment is the interstate highway with light-medium traffic and consistent speed limit. This testing environment allows us to test the controller if it works properly when changing lanes or adjusting the acceleration value. The last assumption is that the passenger should identify when the controller should enter different modes and verify that the transition occurred to ensure the safety.

Design

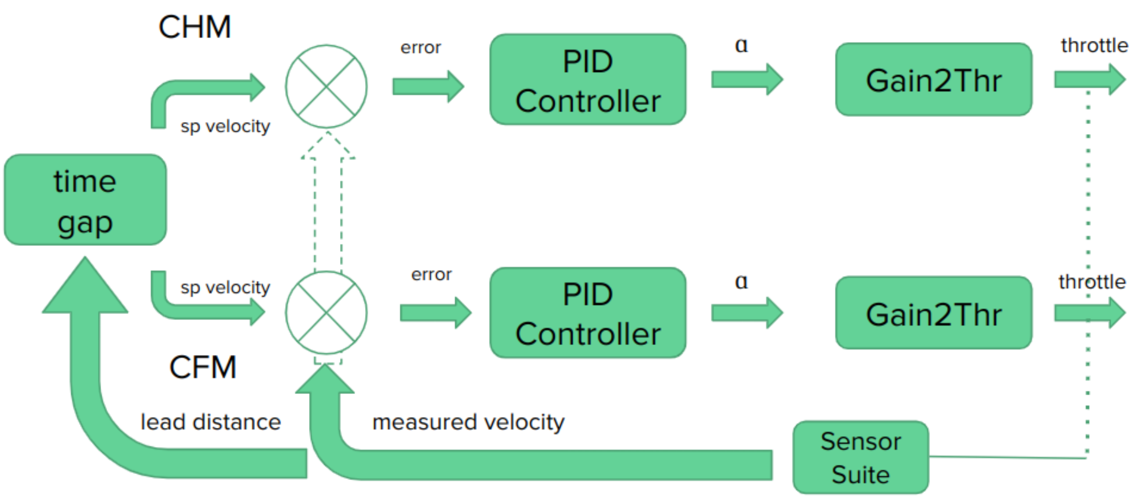

Figure 1: Control Model

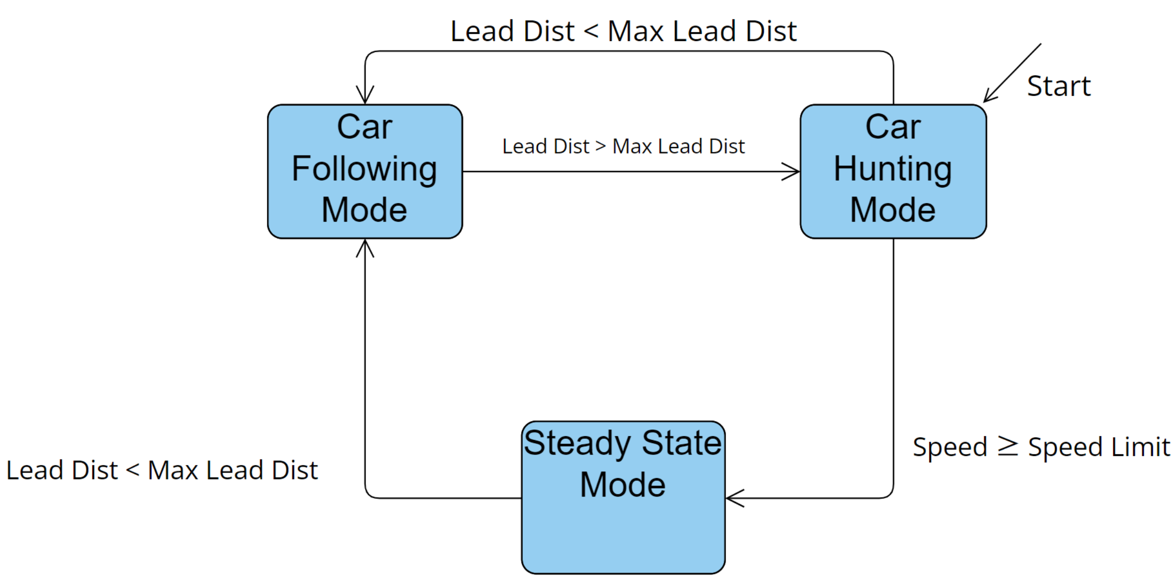

Figure 2: Finite State Machine

Figure 1 above displays the block diagram for the control model of the speed controller. It can be characterized as two PID controllers to represent the Car Following Mode (CFM) and Car Hunting Mode (CHM) respectively (note Figure 2 for State Machine Model). The control flow for the system is as follows. Prior to the initial start up of the controller the user will define the speed limit (Float64 bit) of the roadway, the optimal lead distance (Float64 bit) for the vehicle, and the maximum threshold value (Float64 bit) for the lead distance. After engaging the control system, it will sample data from the car's sensor suite and retrieve the most recent capture of the vehicle's velocity (Float64 bit) and lead distance (Float64 bit) from the vehicle in front of it.

The lead distance will be used by the Time Gap block in order to determine the speed that the vehicle needs to travel in order to maintain the optimal lead distance with the vehicle in front of it. In this block, if the lead distance is greater than or equal to the maximum threshold lead distance then the Time Gap block will pass a velocity value into the Car Hunting Mode's PID controller which is characterized by higher gains to accelerate the vehicle at faster speeds. If the lead distance is less than the maximum threshold lead distance then the Time Gap block will pass a velocity value into the Car Following Mode's PID controller which is characterized by more normalized gains so that the vehicle can properly follow the vehicle in front of it.

After both PID blocks in either mode the architecture is designed to be approximately the same. The gain value is processed in the Gain2Thr block where it is converted into a throttle value that the vehicle will receive in order to accelerate/decelerate. Additionally within this block there will be data processing to check whether the speed that the vehicle is traveling at exceeds the speed limit of the roadway, if so it will then set the gain value to 0, because the vehicle is not allowed to exceed the speed limit. Otherwise it will process the gain for safety and ensure that the vehicle is accelerating in a manner that is safe for the occupants. After this the car will change its state, following the sensors will sample the new velocity and lead distance and repeat the loop.

We assume that the frequency for sampling the sensor suite of the vehicle will be around nominally 60Hz, and that our outputs will occur within a slightly faster response time of perhaps 65Hz. These are estimations of our control systems performance and are subject to change with further testing and calibration of the system.

Metrics For Success

Our project will be evaluated on the target vehicle's ability to:

1. Follow a lead vehicle at a set lead distance.
2. Detect when the space gap has grown too large.
3. Accelerate and either close the gap or maintain the speed limit.
4. Perform the above with safe smooth maneuvers that ensure the passengers are comfortable.

Implementation

The implementation of our controller is divided into three general steps.

Step One:

Time Estimate: 2 weeks.

Team Members Involved: Mainly Ryan Taylor with support from Hugo Son and Daniel Little as needed.

Details: this portion of the implementation is where we will create the three modes for our controller in Simulink. We will ensure that the transitions between modes are handled correctly (e.g. the controller successfully switches from Car Hunting Mode to Car Following Mode, etc.) before designing the appropriate subsystems to handle each mode of operation.

Step Two:

Time Estimate: 1 week.

Team Members Involved: Hugo Son and Daniel Little, with support from Ryan Taylor as needed.

Details: this is the main testing stage for our design. We will simulate our model with ROSbag playback and adjust model parameters to improve the performance of our controller.

Step Three:

Time Estimate: 1 week.

Team Members Involved: All members.

Details: this is the final step before real-world testing. We will conduct more simulations and tests with a focus on ensuring that the controller creates a realistic, smooth driving experience with no sudden accelerations/decelerations and unsafe behavior.

Collaboration

Ryan Taylor will be responsible for the implementation of the controller modes in

Simulink. Daniel Little and Hugo Son will be in charge of developing the simulating environment to test the control code and making model adjustments based on simulation results.

References

Levy, J.I., Buonocore, J.J. & von Stackelberg, K. Evaluation of the public health impacts of traffic congestion: a health risk assessment. *Environ Health* 9, 65 (2010). https://doi.org/10.1186/1476-069X-9-65

Inrix. “INRIX 2022 Global Traffic Scorecard.” *Inrix.com*, INRIX, inrix.com/scorecard/. Accessed 24 Oct. 2023.