

Research Journal

Chris Kreienkamp

ckreienk@email.arizona.edu

Week 2

Monday, June 10

Met with Rahul today to discuss possible directions where we could take the project, but my partner Fish was not present. Rahul summarized our meeting and sent a dense email filled with resources. I read through and took notes on "Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments?" and on "The CAT Vehicle Testbed: A Simulator with hardware in the Loop for Autonomous Vehicle Applications." Tomorrow I hope to read the rest of the papers (4) in the email that concern the FollowerStopper. On Wednesday and Thursday I hope to familiarize myself with ROS, Gazebo, and the Testbed. Over the weekend I hope to focus on learning Simulink and supplementing my knowledge of control systems.

Tuesday, June 11

I read through and took notes on "Dissipation of emergent traffic waves in stop-and-go traffic using a supervisory controller," "Tracking vehicle trajectories and fuel rates in phantom traffic jams: Methodology and data," and "Real-time distance estimation and filtering of vehicle headways for smoothing of traffic waves." The first and third gave in-depth descriptions of the Follower Stopper controller. After reading, it is clear that the parameter in the FollowerStopper configurable quadratic bands can be optimized through more research. The second paper discussed the circular phantom jam experiment, giving a special look on the tracking methods employed to find the instantaneous positions, velocities, accelerations, and fuel rates of all vehicles in the experiment. The third paper focused on the designing of filters to reduce the noise in relative velocity data of the lead vehicles and to calculate the expected separation distance between the AV and the lead vehicle. I came up with a few questions because of the reading: Would it be beneficial to research how animals respond to traffic in a herd? What if we looked at how acceleration/deceleration cause car-sickness or imbalance in the ear when determining the maximum deceleration parameter in the FollowerStopper? Would it be possible to develop a closed-form mathematical model establishing a relationship between safety metrics, sensor frequency, and desired accuracy of the sensors in order to optimize use of more inexpensive sensors?

Wednesday, June 12

I spent most of the day trying to unsuccessfully download MATLAB on my laptop. Otherwise, I have been working through ROS tutorials. I made it through about half of the tutorials on <http://wiki.ros.org/ROS/Tutorials>, and I hope to make it through the rest by tomorrow.

Thursday, June 13

I tried to work with Simulink, but my computer took exceptionally long and would keep freezing in Gazebo. Additionally, I spent some time trying to understand ROS.

Friday, June 14

After spending the day gaining information from students, administrators, and professors about graduate school, I tried to improve upon my understanding of ROS. I worked through the rest of the ROS tutorials that I had started on Wednesday, and I watched some YouTube videos to see alternative projects and demonstrations of use. Though I still struggle conceptually understanding ROS, I will suspend further work on the program until I have a task to complete, as then I will be

able to have a more intentional goal when using the program.

Saturday, June 15

Read through chapters 4-7 of Vehicle Dynamics and Control by Rajesh Rajamani. Chapter 4: Longitudinal Vehicle Dynamics? considered the forces acting on the vehicle in the powertrain. Vehicle dynamic equations were strongly influenced by longitudinal tire forces, aerodynamic drag forces, rolling resistance forces, and gravitational forces. The powertrain consisted of an internal combustion engine, torque converter, transmission, and wheels. Chapter 5: Introduction to Longitudinal Control? provided an introduction to several longitudinal control systems, including standard cruise control, adaptive cruise control, collision avoidance, control of vehicles in platoons, and anti-lock brake systems. Longitudinal control refers to the controlling of the longitudinal motion of the vehicle (e.g. velocity, acceleration, distance from preceding vehicle) through actuation of the throttle and the brakes. In cruise control, there is an upper level controller that determines the desired acceleration and a lower level controller that determines the throttle input required to track the desired acceleration. Chapter 6: Adaptive Cruise Control? explained that the ACC system is an extension of the standard cruise control, equipped with a radar or other sensors that measures the distance to other preceding vehicles on the highway. ACC systems are autonomous, meaning that only on-board sensors are used and there is no communication between vehicles. There are two modes of steady-state operation: 1) speed control and 2) vehicle following (i.e. spacing control). Additionally, the vehicle following control system must satisfy individual vehicle stability and string stability. To define each form of stability, it is necessary to know that spacing error refers to the difference between the actual spacing from the preceding vehicle and the desired inter-vehicle spacing. The ACC control law is said to provide individual vehicle stability if spacing error converges to zero when the preceding vehicle is operating at a constant speed. The string stability of a string of ACC vehicles refers to a property in which spacing errors, during acceleration and deceleration, are guaranteed not to amplify as they propagate towards the tail of the string. Even by this definition and several other definitions, I am still confused by the notion of string stability. Why would string stability matter if in steady state the spacing error is zero? The chapter also proves that the constant time-gap spacing policy rather than the constant spacing should be used to achieve individual vehicle stability and string stability. Chapter 7: Control for Vehicle Platoons? determines that the vehicle following control system in a platoon should also satisfy individual vehicle stability and string stability. The chapter determines that string stability can only be insured with the constant spacing policy if intervehicle communication is used.

Sunday, June 16

I had not yet reviewed my notes or finished my daily journaling, so I created this LaTeX document and reviewed how to use GitHub, as I have never used it before. Because I still have trouble understanding string stability but did not have enough time to read all of the papers emailed to me by Rahul in depth, I skimmed all of them. I might give a detailed account of each one in future days. I am still trying to figure out the best way to read the papers. Over the past week, it would take me on average 3 hours to read a paper because I would take such intense notes in order to help me to stay focused on the material, clarify what was new information, and to provide a source to review if I want to get a general overview of a paper from my own perspective. I hope to discuss with Rahul tomorrow if such time spent is worthwhile or wasted, as I could try to read one paper every hour so that by the end of a week I could have forty papers read. In the past I have preferred

to print out papers in order to mark them up, but the inavailability of a printer and large quantity of pages I would be printing makes this option unreasonable. The benefit of my current note-taking is that it forces me to read the paper very closely, but again, I do not know if this is a waste of time. Should I be reading papers to search for something, to accomplish a goal, or to gain information for the purpose of improving my knowledge of the subject? I think it would be beneficial for Sunday to be a day of review for me, where I look back over every paper I have read during the week and do not look at any new material. This would prepare me for the weekly meeting and give me direction for future work. My foremost goal at the meeting tomorrow is to gain an understanding of how many papers I should be reading, to what depth, and by means of what notation. I hope to give more thought into my daily journals, giving true summaries of the works I read, indications of the things I accomplished, and bibliographies for all of the sources I use.

Week 3

Monday, June 17

Presented the "Week 3" presentation that is available on GitLab. The responses were relatively positive. The presentation revealed that I need to obtain a better understanding a string stability, but after talking with Rahul, I have confirmed that my idea of string stability is correct. ?In a string stable platoon of vehicles, small perturbations will be dissipated as they propagate from one vehicle to another, while in a string unstable platoon small perturbations from equilibrium may amplify as they propagate through the platoon.? One example is that if, in a line of cars, the lead car slows down by 6 mph, the line would be string unstable if the fifth car had to slow down by 25 mph. I hope to recreate the ring-road experiment on Gazebo, and to do so, I need to accurately model how humans drive, so I focused my reading on car-following algorithms. I read three more papers entitled, "Are commercially implemented adaptive cruise control systems string stable?", "Car-following: a historical review", and "Car-following models: fifty years of linear stability analysis - a mathematical perspective". The general take-away was that there are many different car-following models, some linear, some nonlinear, but all imperfect. As one paper even stated, "Our view is that there are probably too many microsimulation models in circulation. (It has become fashionable for every researcher to derive his own model, perhaps because this leads to more publications!)" (Wilson 16) Some difficult things to account for include driver reaction time delay and how the driver will react not only to the car in front him but also to the cars he can see that are several cars ahead. Towards the end of the day, I talked to Rahul, who revealed that my efforts need to be directed toward understanding ROS. For the rest of the week, I will make it my goal to try to recreate the ring-road experiment in Gazebo, pausing my reading of any future papers until I have a deep understanding of the software.

Tuesday, June 18

Worked through ROS tutorials on the ROS Wiki. Detailed notes were taken and recorded in my binder.

Wednesday, June 19

Worked through the Gazebo tutorials on gazebo-sim.org. I also followed a YouTube video entitled, "[Tutorial] Building a Simulated Model for Gazebo and ROS from Scratch (part 1)" to learn more about the file system in Gazebo. For example, in the urdf folder, *.xacro contains the urdf specifications that describe the geometry, visual specifications, links, and joints of a robot.

Thursday, June 20

Worked the tutorials on <https://cps-vo.org/group/CATVehicleTestbed>, a website seemingly produced by Dr. Sprinkle. Through the simulations, I was able to use RViz to visualize laser data in simulation and was able to upload the Hoffmann Controller from Simulink into Gazebo. RViz is a program used to receive and analyze sensor data.

Friday, June 21

Followed the work of Rahul Bhadani on his YouTube page to create a Simulink model to publish a velocity to the cmd_vel topic. Due to computer issues, I was only able to create a constant velocity controller for one vehicle.

Saturday, June 22

Created two more Simulink models to give the car in Gazebo a sine wave velocity and a random velocity. I wrote a MATLAB script to model the FollowerStopper controller, but had problems incorporating it into Simulink. Creating a working Simulink of FollowerStopper and a car-following model will be the first task next week, with hopes of recreating Rahul's "Vehicle follower in Straight Line with a random velocity to the leader vehicle" video on YouTube. It features a lead vehicle that travels at a random velocity while the car behind it follows according to a car-following model.

Sunday, June 23

Met with my group partner, Fish, to create a presentation for the weekly update tomorrow. Afterwards I made sure that my daily journal was up to date. My weekly plan follows. Monday, Fish and I will give our presentation and then I hope to develop the FollowerStopper and one car-following controller to recreate Rahul's video. Tuesday, I hope to simulate the ring-road experiment and an experiment of cars in a straight line. Additionally I hope to create another car-following controller. Wednesday, I hope to create a third car-following controller and begin a report detailing if FollowerStopper is string stable or unstable. Thursday, Friday, and Saturday, if all is going according to plan, I hope to continue the report and begin to try to optimize the parameters in the FollowerStopper controller.

Week 4

Wednesday, June 26

Created a Simulink model consisting of a lead and follower vehicle car plants, smoothUpParams, and FollowerStopper. smoothUpParams adjusts the reference velocity so that it does not cause the vehicle to accelerate too quickly, decelerate too quickly, or exceed the reference velocity. The model creates position and velocity data to be used by a MATLAB script I created to plot the position-time, velocity-time, and x-y graphs of the lead and follower vehicles.

Sunday, June 30

Goals for the week:

1. Scenario One: line of vehicles
 - (a) Lead car has a synthetic velocity from one of the cars in the ring-road experiment
 - (b) Some follower vehicles immediately after with car-following model
 - (c) One vehicle i with FollowerStopper
 - (d) Some more follower vehicles with car-following model
2. Scenario Two: ring-road experiment
 - (a) Model cars in perpetual traffic
 - (b) One vehicle is FollowerStopper, all others are controlled by the car-following model
3. Scenario Three: repeat scenarios one and two, all with car-following model
 - (a) incorporate a second car-following model
 - (b) repeat scenarios one and two with the new car-following model in place of the old
 - (c) repeat scenarios one and two with a mixture of the old and new car-following models
4. Scenario Four: repeat scenarios one and two, all with FollowerStopper
5. Scenario Five: tune the desired velocity
 - (a) Hand-tailored \Rightarrow expert's advice
 - (b) From global information

Week 5

Monday, July 1

- Incorporated a Filtered Derivative block into the carPlant_smootherUp_followerStopper.slx Simulink file to obtain accurate information about the acceleration of the lead and follower vehicles. Changed the Simulink file so that the FollowerStopper controller would publish to a node rather than send the commanded velocity straight to the input of the follower vehicle to fix an algebraic loop issue caused by smoothUpParams.
- Downloaded [The Arizona Ring Experiments Dataset \(ARED\)](#) from Vanderbilt. Chose Experiment E, car 1, as the vehicle off of which to obtain a synthetic lead vehicle velocity because Experiment E features the most oscillatory traffic without intervention by the CAT Vehicle to modify the flow of cars. Information about the other experiments in the dataset can be found in the paper, [Tracking vehicle trajectories and fuel rates in phantom traffic jams: Methodology and data](#).
- Used the data to create a MATLAB script, Vanderbilt_Dataset.m, which extracts the velocity information for car 1 in Experiment E. The carPlant_smootherUp_followerStopper.slx Simulink file was updated so that this velocity would be the input to the lead vehicle and the reference velocity would be the average velocity for car 1 in Experiment E.
- PROBLEM: With the Vanderbilt data, the carPlant_smoothUp_followerStopper.slx Simulink file looks terrible at the beginning. The reference velocity and the command velocity are unstable, causing an incredible acceleration. I attempted to edit the smoothUpParams to limit the maximum acceleration of the reference velocity, and it looked slightly better. The problem I believe is that smoothUpParams, if the reference velocity max_speed is greater than 2 m/s will set the reference velocity r to 2 m/s. FollowerStopper then immediately commands this velocity, which requires an incredible acceleration from 0 to 2 m/s over a short period of time. (solved 7/2)

Tuesday, July 2

- PROBLEM: Fixed the problem from yesterday concerning smoothUpParams prescribing a high reference velocity r . When FollowerStopper is initiated, the AV under control must be going faster than 2 m/s or must be within ω_1 of the leader vehicle. Yesterday, the lead vehicle was 10 m in front, which was too far ahead. Though this seemingly solves the problem, there is additional noise in the middle part of the FollowerStopper command velocity data when following the synthetic velocity lead vehicle. Also, we will not be able to have all cars start within ω_1 during the recreation of the ring-road experiment, so further development is needed to solve the noise issue without editing the relative position. (solved 7/2)
- Edited the MATLAB functions for followerStopper and smoothUpParams and justified all changes in the document "Suggested Changes to followerStopper and smoothUpParams" on GitLab. The changes solve the above problem and the problem from yesterday. There are still some abrupt changes due to the fact that followerStopper must be able to decelerate to

any velocity at any moment for safety's sake, but the acceleration does not exceed the desired maximum acceleration at any point. The AV can now initiate FS at any initial speed and any initial relative distance. An e-mail was sent to Rahul so that he would accept the changes, because either he or another designer of FS wrote the code.

- QUESTION: How does relative position relate to likeliness of a driver to lane change? We want FS to follow close enough to reduce the chances of lane changes, which will cause cars from behind moving around and more lane changes that will cause traffic. This problem is related to choosing the reference velocity.
- QUESTION: What is the actual comfortable acceleration and deceleration? Up until now, we have been using $1.5 \frac{m}{s^2}$ but a more verified value should be used.

Wednesday, July 3

- DOCUMENT: [An ecological Theory of Motion Sickness and Postural Instability](#): Proposes that animals become sick in situations in which they do not possess (or have not yet learned) strategies that are effective for the maintenance of postural stability. This enables the establishment of a link between motion sickness and postural stability. In flight simulators, the operator learns that the optical accelerations do not have consequences for body balance. When the operator leaves the simulator to drive home, the accelerations in the car could give rise to postural instabilities if the operator continues to decouple postural control strategies from the vehicle control strategies. The paper, unfortunately, gives no acceleration values which we are looking for to set as the maximum acceleration. It merely supports the previously suspected hypothesis that expected movements by the AV are less likely to cause motion sickness.
- CREATE STRING INSTABILITY TEST AND TEST FS AND CF1
- CREATE DELAY FOR FS
- CREATE SECOND CAR-FOLLOWING MODEL
- CREATING THE LINE OF VEHICLES
- EDIT CAR PLANT SO IT HAS THE Y THING FIXED
- ORGANIZE GITLAB

Friday, July 5

- Creating graphs using the stringStabilityTest.slx file, which attempts to test if any of the car-following models or if FollowerStopper is string unstable. There are two tests, acceleration steps and deceleration steps. Acceleration steps: the input velocity to the lead vehicle starts at 0 mph, accelerates to 5 mph, stays at 5 mph for 30 seconds, accelerates to increase the speed by 5 mph, stays for 30 seconds, and continues the trend until the speed reaches 35 mph, at which point the lead vehicle decelerates to 0 mph. Deceleration steps: The input

velocity to the lead vehicle starts at 0 mph, accelerates to 35 mph, stays at 35 mph for 30 seconds, decelerates to decrease the speed by 5 mph, stays for 30 seconds, and continues the trend until the speed reaches 0 mph. Comment out everything except the acceleration and deceleration steps boxes and run those first. Then comment out those and uncomment everything else. This will save time on the simulation.

- stringStabilityTest_1: Acceleration steps test for the edited FS controller and CF1. FS is unstable. CF1 is unstable.
- stringStabilityTest_2: Deceleration steps test for the edited FS controller and CF1. FS is unstable. CF1 is unstable.
- stringStabilityTest_3: Acceleration steps test for the original FS controller and CF1. FS is unstable. CF1 is unstable.
- The above experiments suggests that the edits to the original FS did not make FS unstable, but the original code was also string unstable along with other issues. Perhaps FS needs a better reference velocity, which was originally set to 27 m/s \approx 60 mph.
- HAVE NOT YET IMPLEMENTED THIS: Equation 11 in [Real-Time Distance Estimation and Filtering of Vehicle Headways for Smoothing of Traffic in Waves](#) gives the total delay to be

$$\delta = \frac{\delta_f F_s}{F_v} + \delta_r. \quad (1)$$

P. 8 of the same paper reveals the sampling frequencies of the sensors (F_s) and the lead vehicle velocity (F_v) to be the same, giving the total delay δ as the sum of the filter delay (δ_f) and the fixed delay due to friction, actuator dynamics, controller update rate, etc. (δ_r). A camera introduces a delay of $\delta_f = 0.333$ s whereas LiDAR introduces a delay of $\delta_f = 0.133$ s. For a conservative estimate, the delay of the camera is used in conjunction with the fixed delay $\delta_r = 1.0$ s to give a total delay of $\delta = 1.333$ s. Because the time step in Simulink is 0.01, I added a delay to the FollowerStopper controller to react 133 steps after receiving information.