CONTROL STRATEGIES FOR MITIGATING TRAFFIC SHOCK WAVES UTILIZING CONNECTED AND AUTONOMOUS VEHICLES

A Thesis

Presented to

The Faculty of the Department of Electrical and Computer Engineering

California State University, Los Angeles

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Electrical Engineering

By

Joshua Adam Saunders

August 2018

The thesis of Joshua Adam Saunders is approved.

Dr. Shaurya Agarwal, Committee Chair

Dr. Charles W. Liu

Dr. Nancy Warter-Perez

Dr. Fereydoun Daneshgaran, Department Chair

California State University, Los Angeles

August 2018

ABSTRACT

Control Strategies for Mitigating Traffic Shock Waves Utilizing Connected and Autonomous Vehicles

By

Joshua Adam Saunders

In this thesis, an optimal control strategy to mitigate shock waves in traffic streams consisting of connected and autonomous vehicles is developed and compared against a that of a proportional controller. First, the formation of shock-waves is explained with the help of mathematical modeling and numerical simulation. Further, an optimal control problem to mitigate shock waves on a circular track consisting of connected and autonomous vehicles is formulated. The optimal control problem is solved using linear quadratic tracking controller using the variational approach. We use entropy to measure the effectiveness of connected and autonomous vehicles to reduce shock waves (stop-and-go) waves on a circular track, thereby increasing throughput and reducing emissions. An optimal control law is also developed to minimize the error of the headway between vehicles in a platoon.

ACKNOWLEDGMENTS

I would like to thank to L.A. Adamic, N. Glance, and Gallup for the use of their images in the spirit of academic advancement. Without these resources, my points would both less strong and accessible.

This work would not have been possible without the financial support of California State University, Los Angeles and their generous academic grants and the support of the faculty and staff of the Electrical and Computer Engineering Department. I am especially indebted to my professors during my time here who have been supportive of my career goals and who have worked actively to help me to pursue those goals, as well as providing their extensive professional and personal guidance. I would also like to thank my thesis committee members for their input, advice, and interest in this work.

I am grateful to all of those with whom I have had the pleasure to work during this and other related projects. I would especially like to thank Dr. Shaurya Agarwal, the chairman of my committee. As my teacher and mentor, he has taught me more than I could ever give him credit for here. He has shown me, by his example, what a good researcher, engineer (and person) should be.

Nobody has been more important to me in the pursuit of this project than the members of my family. I would like to thank my parents, whose love and guidance are with me in whatever I pursue. They are the ultimate role models. Most importantly, I wish to thank my loving and supportive fiancée, Liliane, who has provided unending understanding, encouragement, and inspiration.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure

CHAPTER 1

Motivation

1.1 Why Traffic is Important

Transportation has played an integral role in human society from the very beginning, allowing humans to colonize different geographical regions, trade amongst different civilizations, and to spread ideas. Initially, humans walked and carried items on their backs and heads. Beasts of burden such as mules or oxen were then used which allowed for more goods to be transported at a higher speed. Then, roads were built and people and goods were carried by wheeled carts. As transportation systems became more complex and advanced, people and goods were able to be transported further and at greater speeds.

Transportation has played an integral role in human society from the very beginning, allowing humans to colonize different geographical regions, trade amongst different civilizations, and to spread ideas. Transport by land started by foot with people walking from place to place carrying goods on their backs and heads. By the early twentieth century, automobiles began to be mass produced in America. Initially they were seen as toys for the rich.

Traffic congestion, air and noise pollution, and safety are common problems that many cities face today. It is estimated that 50% of the total population is living in cities, and by 2050, this percentage is projected to grow over 70%, i.e. over six billion people will live in cities and urban areas. In 2014, U.S. drivers drove over 2 trillion vehicle miles, with congestion costing over 5 billion dollars on average for very large urban areas and over 191 million dollars on average in small urban areas. In 2014, there was a reported 5.9 million accidents by passenger vehicles and the costs and accident rates are increasing on average across the nation [1].

The transportation engineering and planning communities are witnessing the emergence of a new generation of traffic systems, also known as connected and autonomous vehicles (CAVs). Recent advancements in CAVs are expected to transform how people use transportation. There is a growing interest in the research community to exploit the available vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication technologies in CAVs, to improve the efficiency and safety of traffic flow while reducing emissions. It is estimated that integration of CAVs will reduce about 40% of fatal traffic accidents [2]. Aside from passenger and road safety, researchers also envision that CAV technology will reduce congestion and fuel consumption [3].

The ability to communicate between vehicles improves the perception of the driving environment and the quality of driving related decisions. V2V communications provide detailed information about vehicle movements and operation decisions (e.g., speed and acceleration), while V2I communications provide detailed information on road and weather conditions. The CAV is also able to obtain information from adjacent vehicles and predict the traffic stream parameters ahead of them (e.g., shockwave formations). The availability of V2I communications provides information on breakdown formations (e.g., lane closures or accidents), changes in the speed limit, work zone conditions, and additional information [4]. User experiences are important for smoother transition to next generation technologies. Bansal and Kockelman per-formed a national survey to understand the future vehicle preferences related to CAV technology [5]. As noted by the respondents, traffic sign recognition and left turn assist was of no interest, as about 46% of users were not willing to pay for the technologies to be added to their vehicles. However, respondents were very interested in a blind-spot monitoring system. As a result, many manufacturers implemented this technology in their vehicles. About 54.4% of the respondents perceived CAVs as a useful advancement in the near future. Furthermore, 50.4% of the respondents were comfortable with their vehicle transmitting information to other vehicles and 62.3% were willing to trust technology companies. Most users show signs of accepting the changes, and as CAV technology continues to advance, it will provide a positive impact on the future of mobility.

1.2 Trends Towards Automation in the Automotive Industry

Like many other industries, the automotive industry is benefiting from the increasing capabilities and especially shrinking footprint of computers. Processing power is getting cheaper and because of this very powerful computers are able to fit into smaller volumes including the cramped spaces inside of motor vehicles. Not only are computers more powerful, smaller, and cheaper than ever before so are sensors. Lidar, radar, and cameras are relatively cheap, powerful, and have high resolution. Powerful computers and sensors in novel objects is fertile ground for innovation. Large automakers and smaller startups (many funded by large companies and/or investors) are in a race to create the first autonomous vehicle.

1.2.1 Current Capabilities

SAE International breaks down autonomous vehicles into six automation levels. These levels are given in Table 1.1 and are sourced from the NHTSA [6]. The levels of automation range from the human driver being completely in control 100% of the time (Level 0) to an Advanced Driver System (ADS) is fully in control with no human intervention necessary (Level 5). A great example of advancing automation in the automotive industry is Tesla.

Currently, the "Tesla Autopilot" (which is one of the most advanced ADS on the market) is at Level 2 which means that Tesla vehicles are capable of steering within a lane, maintaining a speed relative to other vehicles, assisting lane changes, parking, and summoning to and from garage. Tesla will continue to update the Autopilot on its vehicles (which Tesla will the Enhanced Autopilot), steadily adding new features such as automatic lane keeping, speed matching, lane changing, freeway exiting, parking when near a parking space. Like many other car-makers, Tesla plans to eventually have its cars be self-driving [7, 8].

1.2.2 Research and Testing

There are many companies that are competing in the race to the elusive self-driving car. Some of the biggest names are Uber, Waymo (formerly known as the Google self-driving car project), NVIDIA Corporation, and major car manufacturers such as Honda, GM Cruise LLC, and Tesla [9]. Because of the secretive nature of industrial research it's difficult to determine just how far along each company is towards achieving full autonomy for self-driving cars. However, one way to determine

Table 1.1: SAE International automation levels

Automation Level	Who does what, when
0	The human driver does all the driving.
1	An advanced driver assistance system (ADAS) on the vehicle can sometimes assist the human driver with either steering or braking/accelerating, but not both simultaneously.
2	An ADAS on the vehicle can itself actually control both steering and braking/accelerating simultaneously under some circumstances. The human driver must continue to pay full attention (monitor the driving environment) at all times and perform the rest of the driving task.
3	An Automated Driving System (ADS) on the vehicle can itself perform all aspects of the driving task under some circumstances. In those circumstances, the human driver must be ready to take back control at any time when the ADS requests the human driver to do so. In all other circumstances, the human driver performs the driving task.
4	An ADS on the vehicle can itself perform all driving tasks and monitor the driving environment essentially, do all the driving in certain circumstances. The human need not pay attention in those circumstances.
5	An Automated Driving System (ADS) on the vehicle can do all the driving in all circumstances. The human occupants are just passengers and need never be involved in driving.

how far along companies are towards realizing a fully autonomous (Level 5) car is to examine documents that they are required to release. One such document is the Autonomous Vehicle Disengagement Report that the California Department of Motor Vehicles requires as a part of the licensing requirements for performing field tests on California roads under the Autonomous Vehicle Testing Program.

Autonomous Vehicle Disengagement Reports contain the number of miles driven autonomously on public roads in California per month and disengagement events and related information. During field tests of autonomous vehicles, a driver must be in the driver's seat of the vehicle [10]. A disengagement event is

> "a deactivation of the autonomous mode when a failure of the autonomous technology is detected or when the safe operation of the vehicle requires that the autonomous vehicle test driver disengage the autonomous mode and take immediate manual control of the vehicle, or in the case of driverless vehicles, when the safety of the vehicle, the occupants of the vehicle, or the public requires that the autonomous technology be deactivated [11]."

The information related to this event is what initiated the event (driver or autonomous system), the location of the event, and a description of the event [12]. A very simple method to the progress that industry is making towards Level 5 autonomy is to examine all of the Autonomous Vehicle Disengagement Reports for each company for every year. By doing so, a picture of each company's progress begins to emerge.

It's interesting to note that Tesla is not included in 2017. In that year's report Tesla states that it has not had any disengagement events. This is not because Tesla is

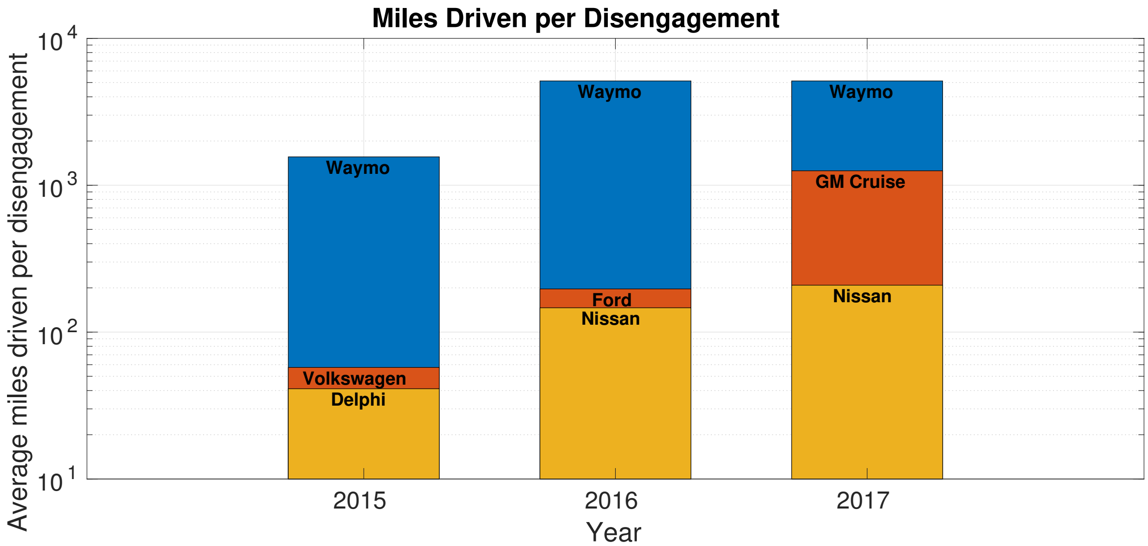

Figure 1.1: Disengagements per year for the top three testers

not performing any autonomous tests, it's just not performing any autonomous tests as defined by California law. Tesla is performing autonomous tests via "simulation, in laboratories, on test tracks, and on public roads in various locations around the world" [13]. In addition, it's testing its control algorithms against the human drivers of its vehicles out in field and thus considering its hundres of thousands of customers' vehicles part of its test fleet. By doing this Tesla has been able to acquire billions of test miles in a very short time. This is accomplished by the vehicles running in "shadow mode" which allows Tesla to collect telemetry data such as braking and acceleration and Autopilot data and then uses this to develop and test its control algorithms safely as none of the algorithms actuate the vehicle [14, 13].

1.2.3 Integration of Artificial Intelligence

According to Chen et al, in much of industry, mediated perception techniques for vision-based autonomous driving systems are used [15]. Mediated perception techniques use subcomponents to recognize traffic data such as other vehicles, lanes, traffic signs and signals, and pedestrians. This data is then combined to create a representation of the traffic scenario such as splines for lane detections and bounding boxes for vehicle detections. An AI-based control system can then take in this representation, determine appropriate trajectories, and actuate the vehicle. Chen et al. describe a few problems with this technique, however.

First, the representation that is created can be too rich, or detailed, meaning that not all of the information in the representation is necessary for the AI-based controller. Second, AI-controllers don't directly use this representation (i.e., it has to be transformed to some other representation that the controller can use) [15]. These transformations from splines and bounding boxes into useful information can create noise and errors that add more uncertainty to the mix. Third, structured details from the traffic scene such as straight edges in buildings can cause false positives for other vehicles.

Another technique is the behavior reflex approach [15]. In this technique, sensory input (e.g., radar, lidar, and images) is mapped to driving actions (e.g., steering angles and accelerations or braking) using neural networks. Although straightforward, this technique has flaws. First, driving conditions are never the same. Even if the same cars are in the same location at the same time of day, people have different moods and intentions and may exhibit different a behavior. Second, the neural network must process an entire image and determine the relevant parts which is difficult for the network.

1.2.4 Obstacles and Limitations

A hurdle for obtaining Level 5 autonomy is that of data. Training neural networks takes a lot of data, which is why we see manufacturers performing extensive field tests. Also, it is necessary to prove that these vehicles are safe and currently the way to prove this is through real-world tests. The RAND Corporation estimates that in order for autonomous vehicles to match the safety rate of human drivers (about 1 fatality in 100 million miles driven) autonomous vehicles need to have in the range of billions to hundreds of billions of test miles driven, which could take hundreds (or even thousands) of years depending on the size of the test fleet and how much better it is desired for the autonomous vehicles to be than human drivers [16]. This is why Tesla is running tests using its customers' vehicles as its own test fleet. The more vehicles it has, the faster Tesla can improve and prove its technology.

Another limitation of the current implementation of autonomy in industry is the lack of coordination of autonomous vehicles and sharing of information. As far publications go, no car companies are talking about if and how their vehicles will share information with infrastructure (V2I) and each other (V2V). Some companies, such as Tesla and Waymo, are discussing about eventually having an autonomous ride-hailing service [17, 18]. However, the lack of sharing information about the vehicles and coordination with other vehicles on the road (including private vehicles) can prove to be problematic for the larger picture. CAVs can increase the throughput and efficiency of roads and highways, but a necessity is the sharing of information between vehicles on the road due to faster response times and the shorter distances (headways) needed between vehicles [19]. If this information is not shared then it will be difficult to glean these efficiencies and much of the improvements of autonomous vehicles will not be realized.

CHAPTER 2

Background and Literature Review

2.1 Traffic Flow Theory

Traffic may be modeled using different levels of models such as macroscopic, microscopic, mesoscopic, and aggregation and disaggregation levels [20]. Macroscopic models describe the aggregate behavior of traffic, similar to how hydrodynamic models describe the aggregate behavior of particles in liquids and gases. Microscopic models on the other hand describe the interactions and reactions of single vehicles with each other similar to multi-particle systems. Mesoscopic models are hybrids of macroscopic and microscopic models. Finally, aggregation and disaggregation is the use of microscopic quantities to describe macroscopic ones (aggregation) and using macroscopic quantities to describe microscopic ones (disaggregation).

The macroscopic quantities that are of interest are the speed (V), density (ρ), and flow (Q) of the vehicles. There are several different definitions of speed, but essentially speed is the distance traveled per unit time. Density is defined as the number of vehicles per unit length of road. Finally, flow is defined as the number of vehicles passing through a certain point in space per unit time.

2.2 Modeling

2.2.1 Macroscopic Models

Much like at a large enough scale a gas or liquid can be described as a continuous entity, traffic can also be described as continuous entity. Ignoring the interactions between individual vehicles on the road, macroscopic models instead focus on the quantities such as flow, density, and velocity of the traffic as whole. This mean that when the different macroscopic model quantities are used, they are in terms of averages, means, or totals.

2.2.1.1 Density

Density, ρ, is defined as the number of vehicles per length of road and can either be for

- a single lane: $\rho_i(x,t)$ where $i = 1, \ldots, n$ and n is the number of lanes
- total density over all lanes of a length of road: $\rho_{tot}(x,t) = \sum_{i=1}^{I} \rho_i(x,t)$
- lane-averaged density: $\rho(x,t) = {}^{\rho_{tot}(x,t)}/_{I}$

2.2.1.2 Velocity

Velocity, V, is defined in terms as a lane-averaged (or, effective) speed and is given by

$$V(x,t) = \sum_{i=1}^{I} \frac{\rho_i(x,t)}{\rho_{tot}(x,t)} V_i(x,t) \tag{2.1}$$

where $V_i(x,t)$ is the local speed of lane i.

2.2.1.3 Traffic Flow Rate

Flow, Q, is defined as the number of vehicles per unit time. Traffic flow rate is the product between density and velocity, as shown in Equation 2.2 and is known as the fundamental relation of traffic flow theory [21]

$$Q(x,y) = \rho(x,t)\, V(x,t)\,. \tag{2.2}$$

Flow can be given in terms of a single lane, all lanes, or as a lane-average shown in Equation 2.2. Many times the flow in terms of all lanes will be used as the conservation of vehicles holds only for the total density, ρ_{tot}, and not for each lane individually as vehicles can change lanes after all. It should be noted that this relation holds for all types of macroscopic models, but not for all types of roads. In other words, the geometry of the road matters but not the model. For example, on-ramps and off-ramps and a change in the number of lanes can allow cars to enter and exit a roadway which require Equation 2.2 to be modified.

2.2.2 Microscopic Models

Microscopic models differ from macroscopic models in that microscopic models are concerned with interactions between vehicles and the reactions and behavior of individual drivers. Many first-order models have the following typical, kinematic model variables for each i^{th} vehicle

- l_i: length of the vehicle
- x_i: position
- v_i: velocity
- a_i: acceleration
- h_i: headway

Not all microscopic models include the headway (bumper-to-bumper distance of the

i^{th} and $i+1^{th}$vehicles), h_i, but it is used enough to point it out. It is important to note that these quantities are not aggregated ones, but are rather for each individual vehicle. Also, because it is not possible to predict human behavior therefore stochastic microscopic models are frequently used [22]. Because these quantities (position, velocity, acceleration, etc.) are known for every vehicle on the road at all times, it is possible to model effects such as economic costs and pollution [22].

2.2.3 Mesoscopic Models

Mesoscopic models bridge the gap between macroscopic and microscopic models. They can be used to relate microscopic parameters to macroscopic ones such as lane-average speed, density, and flow.

2.2.4 Fundamental Diagrams

By assuming that traffic flow rates do not change along the road and through time and all vehicles are the same, then we can use a simple model that does not depend on time or location [21]. Using traffic data that can be obtained using aerial photography, cameras, pneumatic tubes, or other methods plots of the density, velocity, and traffic flow rate can be obtained. These plots relate density and flow rate, velocity and flow rate, and velocity and density. An example using Greenshields' method is shown in Figure 2.1[23].

Much information about the condition of traffic can be obtained from these plots. For example, from the speed-density relation plot we can see that at the lowest possible density ($\rho = 0$) traffic is moving at the highest possible speed and as the traffic density increases the speed of the traffic decreases. In the flow-speed relation

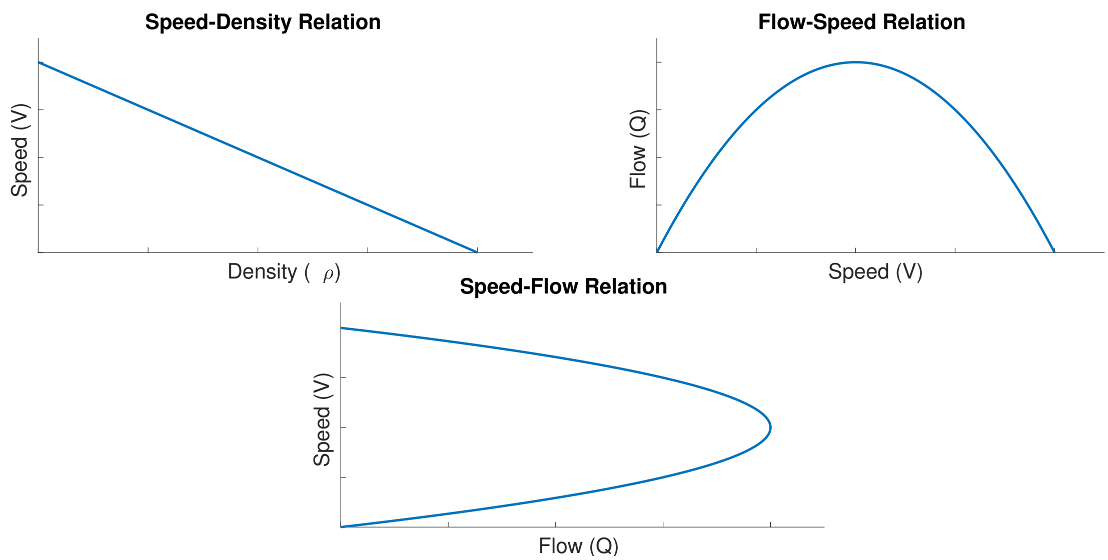

Figure 2.1: Greenshields' fundamental diagrams

plot, it's shown that at the highest speeds the flow will be the lowest and at the lowest speeds traffic flow is also at the lowest value. This is because when the traffic speed is the highest, the density is the lowest, which in turn makes the traffic flow rate low. The second part makes sense, if the traffic speed is low then not many cars are moving. What's a bit more subtle is the peak of the flow-speed relation plot. This is the sweet spot of traffic flow in which we get maximum flow and, as will be shown in Section 3.4, is the boundary between traffic that is in free flow and traffic that is in congestion.

2.2.5 Traffic Shock Waves

A more mathematical treatment of traffic shock waves will be given in Chapter 3.4, however, traffic shock waves can be intuitively understood through the straightforward idea of positive feedback [24]. In general, drivers try to keep a safe distance between themselves and the vehicle ahead of them. If they drive at velocity v and have a response T, then the minimum safe separation distance, d_{min}, is defined by

$$d_{\text{min}} = Tv \tag{2.3}$$

If all drivers are driving at a safe distance from one another, then the density, ρ, is defined by

$$\rho = \frac{1}{\ell + Tv} \tag{2.4}$$

and the throughput, r, is given by

$$r = \frac{v}{\ell + Tv}. \tag{2.5}$$

Instabilities, in the form of waves in density and speed, grow in amplitude and may grow until the speed of the vehicles approach zero [24]. This phenomena can understood through positive feedback. Lower vehicle speeds cause higher traffic densities, which in turn cause lower speeds. On the other hand, higher speeds cause lower traffic densities, which cause higher speeds and so on (of course taking into account speed limits and safety considerations). Therefore, we can see that the system is amplifying the characteristics of traffic (vehicle speed and traffic density) rather than dampening them. This is characteristic of a system with positive feedback. In addition to the system having positive feedback, after a certain density, drivers braking can cause the vehicles behind them to brake, propagating the wave backwards through the roadway.

Horn states that the root of the problem stems from the drivers' feedback and the flow of information. Drivers tend to perform actions that benefit themselves, but tend to be detrimental to other drivers. The flow of information on the road from vehicle to vehicle is unidirectional from the front vehicle to the back. Each vehicle adjusts its acceleration based on the relative position and velocity of the vehicle immediately ahead of it. The deceleration coupled with the backwards, unidirectional flow of information creates an effect that is similar to a shock wave is which also moves backwards through traffic [24].

2.3 Vehicle Platoons

A vehicle platoon consists of a string of vehicles that travel along the road, acting as one single unit, with each car following one another closely at normal highway speeds, as shown in Figure 2.2 [25, 26]. The first car is the designated as the leader and the subsequent cars are followers. Platoons are one strategy to coordinate multiple vehicles together on a road. Platoons can be a heterogeneous mixture of vehicles composed of differenty types of vehicles (sedans, trucks, etc.) and mixed human driver and different levels of autonomous vehicles. In order for platoons to be practical, vehicles need to be able to enter and exit as needed and to drive as close as possible to one another. If the vehicles in the platoon are at the correct distance from one another then there can be cost savings due to aerodynamic efficiencies.

Zabat et al showed that when vehicles in a platoon are close to each other, e.g., race cars drafting during a NASCAR race, then the vehicles are, aerodynamically, one vehicle and there is a reduction of aerodynamic drag and therefore cost savings on

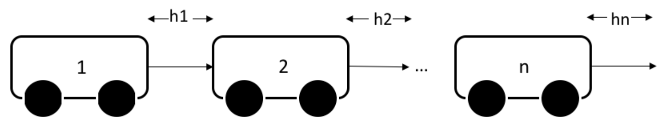

Figure 2.2: An n-vehicle platoon formation

fuel [27]. Aside from cost and fuel benefits, vehicle platoons can increase traffic flow rate. Intuitively, it makes sense why vehicles in a platoon would increase the traffic flow rate. If the vehilces in the platoon are tightly packed then there are more cars are per unit length of road which increases the density and the traffic flow rate, for the same given speed of traffic. However, vehicles moving down a highway at highspeed at close distances is a situation that has potentail for an accident.

2.4 Connected and Autonomous Vehicles

Autonomous vehicles benefit the passengers of each individual vehicle, but not necessarily the rest of the vehicles on the road. As discussed in 2.2.5, traffic shock waves are created in part by the lack of traffic information upstream on a road. A means to counteract this is to actively share information between vehicles (V2V), infrastructure (V2I), and other devices (V2X) effectively coupling the vehicles via a network connection. By vehicles sharing information such as speed, acceleration, and location and infrastructure relaying information such as road and weather conditions [4].

Traffic congestion is caused by a cascade of driver decelerations down a roadway which, in turn, causes an increase in density of vehicles causing more driver decelerations, and so on. These cascaded decelerations cause a shock wave to move backwards through traffic that can be sustained even after the cause of the shock wave has disappeared. By sharing information throughout and coordinating vehicles within a platoon, these shock waves can be reduced. This increases highway capacity by a factor of 2 or 3, relieves congestion, reduces travel times, and decreases fuel consumption [25].

2.4.1 Vehicle-to-Vehicle Communication

2.4.2 Vehicle-to-Infrastructure Communication

2.4.3 Vehicle-to-Device Communication

2.5 Literature Review

Traffic research with vehicle platoons extends back to at least the 1950s with major development taking off in the 1990s a [28, 29]. Currently, there has been an interest in research regarding modeling lane changing, control of platoons at intersections, and string stability of platoons. Much work with vehicle platoons has been completed in recent years. There are two different types of methods of coordinating vehicles that have been proposed in current research: centralized and decentralized coordination. In decentralized approaches, information is shared from vehicle to vehicle and in centralized approaches information is coordinated through a global coordinator. For example, in highway merging problems decentralized approaches handle each vehicle as an autonomous entity that aims to maximize efficiency. On the other hand, in centralized approaches vehicles are controlled by a global agent [30].

The efficiency of traffic flow was quantified by Stern et. al. Experiments were conducted on a circular track as it represents an infinitely long straight track making it simple to identify stop and go waves (i.e., shock waves)[31]. In the experiments, consisting of more than twenty non-autonomous vehicles, uniform speed could not be maintained. With the introduction of an autonomous vehicle at the front of the formation the vehicles reached the uniform, desired speed.

Kachroo et. al. used Mean Field Games and Radon measure with connected and normal vehicles to analyze hetergeneous traffic on a circular track [32]. The connected vehicles were treated as discrete agents which could be used to control the overall traffic streams and entropy was used as a measure of traffic flow efficiency. Their analysis included the effect of various penetration levels of CAVs in traffic stream, their placement, and various sizes of circular track. It was shown that the use of connected vehicles improved the efficiency by lowering the entropy.

CHAPTER 3

Mathematical Modeling of Traffic Flow

Modeling traffic can happen at many different levels such as at the macroscopic and microscopic levels each with it's own strengths and weaknesses. For example, macroscopic models have the advantage of seeing the big picture of traffic and not taking the interaction of vehicles into concern. The disadvantage is that different driving behaviors cannot be modeled so much information is lost. On the other hand, microscopic models do model the interactions of vehicles which allows different driver behaviors to be included in simulations. Also, because they model each individual vehicle on a road, microscopic models have access to the parameters of the vehicles such as acceleration, noise levels, and pollution levels and can therefore give a different view of traffic. However, depending on the number of vehicles that are being simulated microscopic models can require significant computing resources whereas macroscopic models would not.

3.1 Macroscopic Model

Here, traffic will be modeled using a macroscopic model using a hydrodynamic flow-density relation. Before the model is developed it is important define the parameters that will be used, which are given in Table 3.1. It is also important to note that each of the parameters in Table 3.1 have different forms such as average, effective, and total.

Table 3.1: Model parameters for macroscopic model of traffic flow

Parameter	Description
$\rho\,(x,t)$	Traffic density
$Q\,(x,t)$	Traffic flow
$V\,(x,t)$	Lane-averaged, or effective speed
I	Number of lanes
n	Number of vehicles

3.1.1 Continuity Equation

In physics, the continuity equation is a consequence of the conservation of matter that relates the density and velocity of a fluid [33]. Its analog in electromagnetism is the conservation of electric charge [33]. In traffic flow theory, the conservation of vehicles is the analog to the conservation of matter and conservation of charge. On a certain length of road, the number of vehicles is conserved and given by

$$n\,(t) = \int_{x}^{x+\Delta x} \rho_{tot}\,(x',t)\,dx' \approx \rho_{tot}\,(x,t)\,\Delta x. \tag{3.1}$$

For a road section with no on-ramps or off-ramps and no change in number of lanes (i.e., a homogeneous road section) of length Δx, the continuity equation can be derived as follows [20]

$$\begin{aligned} \frac{dn}{dt} &= Q_{in}\,(x,t) - Q_{out}\,(x,t) \\ &= Q_{tot}\,(x,t) - Q_{tot}\,(x+\Delta x,t) \\ &= \frac{\partial}{\partial t}\left(\rho_{tot}\,\Delta x\right). \end{aligned} \tag{3.2}$$

Rewriting Equation 3.2 gives

$$\begin{aligned} \frac{\partial \rho_{tot}}{\partial t} &= \frac{1}{\Delta x}\frac{dn}{dt} \\ &= -\left(\frac{Q_{tot}\left(x+\Delta x,t\right)-Q_{tot}\left(x,t\right)}{\Delta x}\right) \end{aligned} \tag{3.3}$$

and

$$\begin{aligned} \lim_{\Delta x \to 0} \frac{\partial \rho_{tot}}{\partial t} &= -\frac{\partial Q_{tot}\left(x,t\right)}{\partial x} \\ \frac{\partial \rho_{tot}}{\partial t} &\approx -\frac{\partial Q_{tot}\left(x,t\right)}{\partial x}. \end{aligned} \tag{3.4}$$

Finally, rewriting Equation 3.4 we get

$$\frac{\partial \rho_{tot}\left(x,t\right)}{\partial t} + \frac{\partial Q_{tot}\left(x,t\right)}{\partial x} = 0. \tag{3.5}$$

We can use Equation 2.2 (the hydrodynamic flow relation) to rewrite Equation 3.5 (and using the fact that $\frac{\partial Q_{tot}}{\partial x} = \frac{\partial \rho_{tot} V}{\partial x}$) and arrive at the continuity equation

$$\boxed{\frac{\partial \rho_{tot}}{\partial t} + \frac{\partial \rho_{tot} V}{\partial x} = 0} \tag{3.6}$$

or

$$\boxed{\frac{\partial \rho}{\partial t} + \frac{\partial \rho V}{\partial x} = 0} \tag{3.7}$$

3.1.2 Lighthill-Whitham-Richards Models

Lighthill-Whitham-Richards (LWR) models are a class first-order macroscopic models, i.e., they don't include acceleration, and they are all the same except for the form of their respective fundamental diagrams and mathematical representation which is determined by the modeling of the flow and speed only [20]. LWR models give flow and velocity in a functional form

$$\begin{aligned} Q\left(x,t\right) =& Q_e\left(\rho\left(x,t\right)\right) \\ V\left(x,t\right) =& V_e\left(\rho\left(x,t\right)\right). \end{aligned} \tag{3.8}$$

Equation 3.8 assumes that local flow (Q_i) or speed (V_i) are always in equilibrium with respect to the actual density, ρ. It also assumes that the flow and local velocity changes instantaneously to follow the density.

Using the continuity equation, Equation 3.5, the LWR model can be defined as

$$\boxed{\frac{\partial\rho}{\partial t} + \frac{dQ_e\left(\rho\right)}{d\rho}\frac{\partial\rho}{\partial x} = 0} \tag{3.9}$$

if we recognize that $\frac{\partial Q_e(\rho)}{\partial x} = \frac{dQ_e(\rho)}{d\rho}\frac{\partial\rho}{\partial x}$ due to the chain rule. Equation 3.9 can also be written as

$$\boxed{\frac{\partial\rho}{\partial t} + \left(V_e + \rho\frac{dV_e}{d\rho}\right)\frac{\partial\rho}{\partial x} = 0}. \tag{3.10}$$

which is the LWR model.

3.2 Microscopic Models

3.2.1 Human Driver Model

To mimic the traffic shock waves to be controlled, a human driver model is used. The dynamics of the human driver given by Jin et al and Orosz et al is

$$
\begin{aligned}
\dot{h}_i(t) &= v_{i+1}(t) - v_i(t) \\
\dot{v}_i(t) &= \alpha_i \left(V_i\left(h_i(t)\right) - v_i(t)\right) \\
&\quad + \beta_i \left(v_{i+1}(t) - v_i(t)\right)
\end{aligned} \tag{3.11}
$$

where $V_i(h)$ is the distance-dependent velocity, or range policy, given by

$$
V_i(h) = \begin{cases} 0 & \text{if } 0 \leq h_i \leq h_{stop} \\ \frac{v_{max}}{2} A\left(h_i(t)\right) & \text{if } h_{stop} < h_i < h_{go} \\ v_{max} & \text{if } h_{go} \leq h_i \end{cases} \tag{3.12}
$$

where $A\left(h_i(t)\right) = \left(1 - \cos\left(\pi \frac{h_i - h_{stop}}{h_{go} - h_{stop}}\right)\right)$ [34, 35].

In the $v_i(t)$ term of 3.11, the α_i term can be thought of how much the human driver prefers to match the speed given by 3.12 and the β_i term can be thought of how much the human driver prefers to match the vehicle immediately ahead of it.

3.2.2 Intelligent Driver Model

Treiber and Kesting describe the Intelligent Driver Model (IDM) as "probably the simplest complete and accident-free model producing acceleration profiles and a plausible behavior in essentially all single-lane traffic situations" [20]. The IDM defines a safe headway, s_0, and time gap, T, respective to the lead vehicle, has soft braking maneuvers, and smooth transitions between driving modes. The mathemat-

Table 3.2: Model parameters for the Intelligent Driver Model

Parameter	Description
v_0	Desired speed
T	Time gap
s_0	Desired headway
δ	Acceleration exponent
a	Acceleration
b	Comfortable deceleration

ical model for the IDM is

$$\begin{aligned} \dot{x}_i &= v_i \\ \dot{v}_i &= a\left[1-\left(\frac{v}{v_0}\right)^{\delta}-\left(\frac{s^*(v,\Delta v)}{s}\right)^2\right] \\ s^*(v,\Delta v) &= s_0+max\left(0, vT+\frac{v\Delta v}{2\sqrt{ab}}\right). \end{aligned} \tag{3.13}$$

Treiber and Kesting break s^* into two components: the equilibrium term $s_0 + vT$ and the dynamical term $\frac{v\Delta v}{2\sqrt{ab}}$. The dynamical term takes care of the situation when the vehicle is approaching the lead vehicle

Figures 5.1 and 5.2 show the headways and speeds of three vehicles during simulated traffic shock waves, respectively.

3.2.3 Cellular Automata

Cellular automata are simple objects that have cells, grid, and neighborhood [36]. Cellular automata evolve, or update, by a simple set of rules. Each generation of the cellular automaton updates its state by examining the previous state (generation) and using a simple rule, each cell in the automaton is updated. Each cell represents a position on the road and a 0 represents the lack of a vehicle and a 1 represents a vehicle. A simplistic model of traffic using a cellular automaton is Rule 184 [20].

Rules are numbered 0 to 255 and are transformed into binary then a simple truth table is used to define the evolution of the automaton. In the new generation, each cell examines its and its neighbors' previous states and, based on the rule's truth table, is either occupied, denoted by 1, or empty, denoted by 0. The rule can be expressed as a truth table for Rule 184 is given in Table 3.3.

Table 3.3: Truth table for Rule 184

Previous states	New state
000	0
001	0
010	0
011	1
100	1
101	1
110	0
111	1

Using the simple rule set above with a random initial state, a surprisingly complex simulation of traffic is created and shown in Figure 3.1. It is important to note that each row in the figure is one generation of the traffic simulation. Therefore, one row after another is another time step in the simulation and it can be seen that one row to the next has the vehicles moving from left to right.

Figure 3.1: Cellular automaton model of traffic simulation using Rule 184

3.2.4 Connected Cruise Control Model

A Connected Cruise Control (CCC) vehicle is a vehicle in which information is transmitted and shared between other vehicles. Jin et al gave the dynamics for a CCC vehicle, which is shown in Equation 3.14 [37]. In the CCC algorithm, information such as vehicle headway and velocity is shared between all vehicles via a V2V communication, V2I communication, or both. The dynamics of the CCC vehicles are given by

$$\begin{aligned} \dot{h}_i &= v_{i+1}(t) - v_i(t) \\ \dot{v}_i &= u_i(t) \end{aligned} \tag{3.14}$$

where h_i is the headway (bumper-to-bumper distance) between the i^{th} and $i+1^{th}$ vehicles, u_i is the control action [34].

In the CCC algorithm, a platoon of heterogeneous vehicles is formed and vehicle information is shared between the connected vehicles. It is not necessary that every vehicle in the platoon be a CCC vehicle and human drivers are allowed [37]. This is a very flexible control scheme as it allows different types of information, although relying on acceleration data, and allows for a heterogeneous mix of human-drivers and autonomous vehicles and a mix of connected and non-connected vehicles as well.

3.3 Fundamental Diagrams

The fundamental diagrams are a means to convey traffic states graphically.

Show fundamental diagram...

3.4 Traffic Shock Waves

The formation of traffic shock waves can mostly be explained using the LWR model. Let's define the initial density of the traffic flow to be $\rho(x, 0) = \rho_0(x)$ where $\rho(x, t) = \rho_0(x - \widetilde{c}t)$ and $\widetilde{c}$ is the propagation velocity of the density waves. We can see that $\frac{\partial \rho}{\partial t} = -\widetilde{c}\rho_0'(x - \widetilde{c}t)$ and $\frac{\partial \rho}{\partial x} = \rho_0'(x - \widetilde{c}t)$ which, when combined with Equation 3.9 (the LWR model) leads to

$$-\widetilde{c}\,\rho_0'(x - \widetilde{c}t) + \frac{dQ_e}{d\rho}\rho_0'(x - \widetilde{c}t) = 0,$$

or

$$\boxed{\widetilde{c}(\rho) = \frac{dQ_e}{d\rho} = \frac{d(\rho V_e(\rho))}{d\rho}}. \tag{3.15}$$

Equation 3.15 shows that density waves propagate at a velocity dependent on the change in the flow of traffic. This important result will be used to help explain the formation of traffic shock waves later. If we examine the density wave propagation from the point of view of driver we get

$$\begin{aligned}\widetilde{c}_{\text{rel}}(\rho) &= \widetilde{c}(\rho) - V \\ &= \widetilde{c}(\rho) - V_e(\rho) \\ &= \rho V_e'(\rho)\end{aligned} \tag{3.16}$$

with $V_e'(\rho) < 0$, and thus, from the driver's perspective, when $\widetilde{c}_{\text{rel}}(\rho) \leq 0$ the density variations propagate backwards.

Insert image for use in derivation below

For a single-lane highway, let $n = \rho_1 x_{12} + \rho_2 (L - x_{12})$ be the number of vehicles passing through a sufficiently small section of road. Then,

$$\begin{aligned}\frac{dn}{dt} &= \frac{d(\rho_1 x_{12})}{dt} + \frac{d(\rho_2 (L - x_{12}))}{dt} \\ &= \rho_1 \frac{dx_{12}}{dt} - \rho_2 \frac{dx_{12}}{dt} \\ &= (\rho_1 - \rho_2) c_{12} \\ &= Q_1 - Q_2 \\ &= Q_e(\rho_1) - Q_e(\rho_2) \\ \therefore \quad (\rho_1 - \rho_2) c_{12} &= Q_e(\rho_1) - Q_e(\rho_2)\end{aligned}$$

where $c_{12} = \frac{dx_{12}}{dt}$. Finally, we arrive at

$$\boxed{c_{12} = \frac{Q_e(\rho_1) - Q_e(\rho_2)}{\rho_1 - \rho_2}}. \tag{3.17}$$

A great property of the LWR model is that velocities can be determined directly from the fundamental diagram.

- $\widetilde{c}(\rho) = Q_e'(\rho)$: propagation velocity density variations (slope of the fundamental diagram)
- c_{12}: propagation velocity of shock wave fronts (slope of secant line of connecting points of the fundamental diagram)
- $V_e(\rho) = \frac{Q_e(\rho)}{\rho}$: vehicle speed (slope of secant line from origin to corresponding point on fundamental diagram)

CHAPTER 4

Control Methods

4.1 Classical Versus Optimal

Classical control theory concerns itself with the instantaneous control of a system (plant) within design parameters such as rise time, overshoot, etc. Frequency response methods such as Bode plots and root locus are the bread and butter of designing classical control systems [38]. Although these control systems do satisfy the design parameters they are by no means optimal. In fact, the controllers cannot see what's ahead of them or anticipate what's coming next as far as state and control action is concerned. This is where optimal controllers come into play.

Optimal controllers have a long and storied history involving some of the greatest minds of mathematics, physics, and engineering spanning over three centuries and ignoring arbitrary national borders [39]. Optimal control provides a guarantee of optimality through well-developed and rigorously proven theorems. Using a performance measure (or cost functional) engineers are able to give preference to certain behaviors of the controller and to shape and mold it into giving many different trajectories and thus behaviors.

4.2 Proportional, Integral, Derivative Control

4.3 Pontryagin Minimization

4.4 Linear Quadratic Tracking

CHAPTER 5

Problem Formulation

The goal is to remove the shock waves created by three human drivers on a circular track with a radius of 70 meters. In order to do so, we must find the optimal control trajectories that will move the vehicles along the optimal state trajectories so that there is the suppression of the shock waves, which means that the system is stable when every vehicle in the platoon approaches the same constant velocity [37].

To model the human drivers, the IDM is used and the resulting headways and velocities of the human drivers are given in Figure 5.1 and Figure 5.2, respectively. The vehicles are driving on a circular track, as shown in Figure 5.3.

The headways in Figure 5.1 and the velocities in Figure 5.2are oscillating. Because the velocities are not approaching a constant value, the system is unstable and traffic shock waves are occuring.

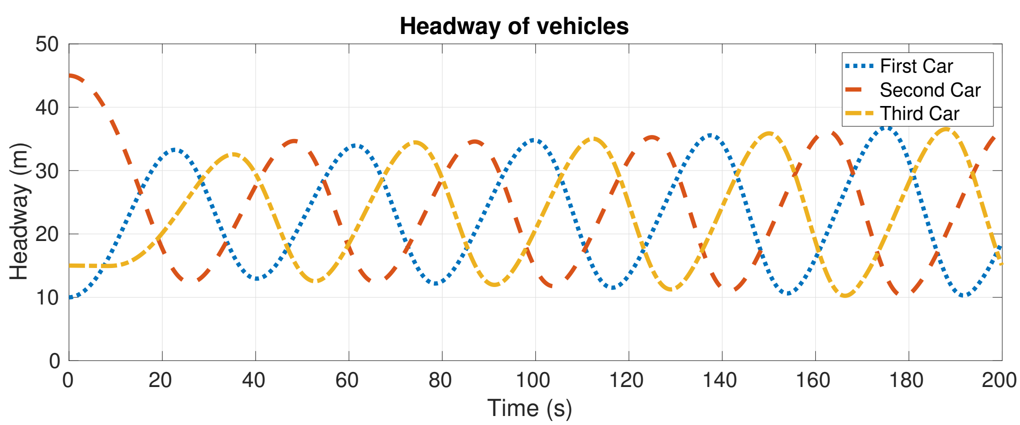

Figure 5.1: Headways of human drivers using the IDM

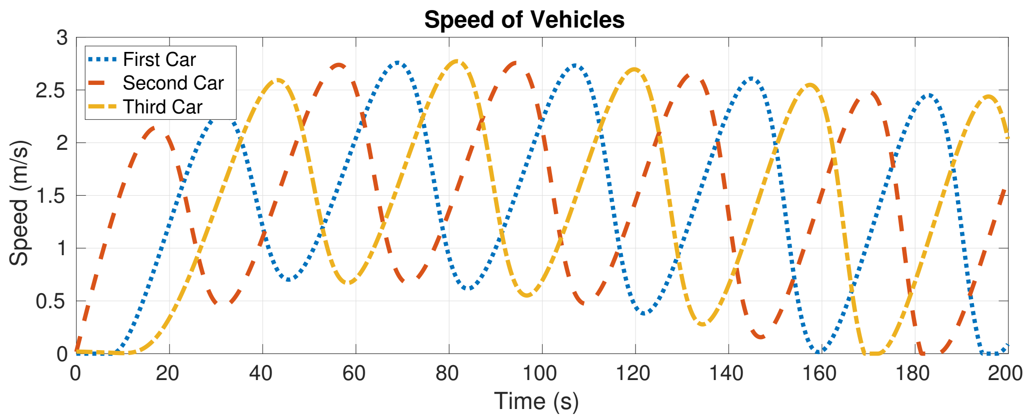

Figure 5.2: Velocities of human drivers using the IDM

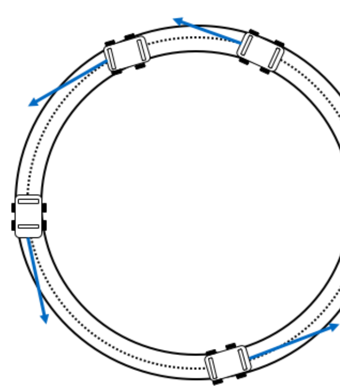

Figure 5.3: Circular slot car race track with five vehicles

5.1 Dynamics

Consider the system shown in Figure 5.3. A chain of 3 vehicles, i.e., a platoon of vehicles, traveling on a circular track. The vehicles implement a Connected Cruise Control (CCC) algorithm given Jin et al [34]. To simplify the model, only vehicles with identical characteristics (homogeneous) are considered. The dynamics of the CCC vehicles are given by Equation 3.14. Each vehicle starts from rest and it is desired to have each vehicle attain its safe headway, h_{safe}, and the maximum possible speed (the speed limit), v_{max}, within time T. Therefore, the initial conditions for the system are given in Equation

$$\begin{aligned} h_i(0) &= h_0^i \\ v_i(0) &= 0 \end{aligned} . \tag{5.1}$$

5.2 Constraints

The system is composed of vehicles that have real, physical constraints. These constraints are both due to the nature of traffic on a road and the physical limits of vehicles and their drivers. Vehicles on the road cannot drive backwards and must not go above the speed limit. Vehicles have a limit on how much acceleration they can produce and much deceleration their brakes provide. Finally, drivers cannot handle large accelerations or decelerations as these may cause injury.

Therefore, the constraints are given such that the control action must be between some predetermined values U_{min} (the maximum deceleration) and U_{max} (the maximum acceleration) such that U_{min} is a negative real number and U_{max} is a positive real number and are physically realistic for a vehicle and are tolerable to a human driver. The constraints are given in Equation 5.2

$$\begin{aligned} 0 \leq & v_i\left(t\right) \leq v_{max} \\ U_{min} \leq & u\left(t\right) \leq U_{max}. \end{aligned} \tag{5.2}$$

5.3 Cost Functional

The cost function is given in Equation 5.3

$$J = \frac{1}{2}\int_0^T \sum_{i=1}^{n} \left(e_i^2\left(t\right) + r_i u_i^2\left(t\right)\right)\, dt \tag{5.3}$$

where $e_i\left(t\right)$ is the error function and r_i is the weighting constant such that $r_i > 0$. The cost function is designed such that the error and control action are minimized resulting in lower error and smooth accelerations.

5.4 Boundary Conditions

CHAPTER 6

PID Control

6.1 Description

6.2 Simulation and Results

CHAPTER 7

Optimal Control

The problem will be solved using optimal control, namely a linear quadratic tracking (LQT) controller that will be developed using a variational approach, as described by Kirk [40]. The reason that an LQT controller is used is twofold. The problem is perfectly suited for it because the dynamics are linear, the cost functional is quadratic, and there is a desired state trajectory that is desired to be followed, or tracked.

7.1 Description

7.2 Simulation and Results

CHAPTER 8

Conclusion

The overriding purpose of this study was to provide an overview of social media information spread theory, modeling, and analysis, as well as provide some examples techniques on how to control information spread in various scenarios. Multiple mathematical models were proposed to better model social media information spread compared to the spreading of information through traditional channels.

The need and importance of studying information spread over social media was established through the use of modern examples, including political campaigning and fake news. A high level background of information categories was presented, along with the important roles played by ignorants, spreaders, and stiflers within a population as it experiences new information. Popular theories on social networking, particularly with respect to online social media groups, were summarized and some of the widely used modern online social networks were identified.

The groundwork for the technical and objective study of social networks was given, including the classifications of relationship types and the qualities that describe them, such as reciprocity, balance, and the presence or absence of homophily. An overview of the technical aspects of social network structure was exhibited, including mathematical formulas and qualitative descriptions and examples. These technical concepts included density, strength of ties, centrality, distance, cohesion, the adjacency matrix and more.

Several models were expressed, described, and simulated to demonstrate their usage in a social media information spread context. Deterministic models included various common epidemiology-based information spread models involving ignorant, spreader, and recovered classes, particularly the popular Maki-Thomson model for rumor spread. Due to the growing influence of online social media networks on the spread of information, several new models were proposed to account for differences between traditional information spread and that of modern digital social media. Stochastic models were also presented briefly, explaining the need for stochastic modeling in real-life information spread systems as well as providing some stochastic models based on the previously discussed deterministic models. Aside from traditional mathematical epidemiology based models, three influential social marketing models were summarized and framed in the context of information spread applications. Again, shortcomings with these models when applied to online social media information spread prompted the proposal of a new model in order to describe online social craze phenomenons.

Two scenarios were created in order to explore the potential of creating a social craze using the proposed model and of applying the idea of herd immunity toward a potential fake news outbreak. For both of these scenarios, dynamics and cost functions were established and the Pontryagin minimization principle was applied to achieve optimal control actions. The scenarios were simulated using MATLAB under a variety of different parameters to demonstrate how changes influence the evolution and control of the systems. Potential real-world examples behind the parameter choices were also discussed.

Future work in online social media information spread would include testing of the proposed models using data from Google, Twitter, or Facebook to track the rise and decay of carefully chosen tag words and images. This data can be used to determine general parameters for similar information spreading systems to hopefully create practical and usable online information spread predictions.

REFERENCES

[1] B. of Transportation Statistics, "Transportation accidents by mode," [Online; accessed 29- Apr- 2018]. [Online]. Available: https://www.bts.gov/content/transportation-accidents-mode. 1.1

[2] T. Li and K. M. Kockelman, "Valuing the safety benefits of connected and automated vehicle technologies," in *Transportation Research Board 95th Annual Meeting*, 2016. 1.1

[3] D. J. Fagnant and K. Kockelman, "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations," *Transportation Research Part A: Policy and Practice*, vol. 77, pp. 167–181, 2015. 1.1

[4] A. Talebpour and H. S. Mahmassani, "Influence of connected and autonomous vehicles on traffic flow stability and throughput," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 143–163, 2016. 1.1, 2.4

[5] P. Bansal and K. M. Kockelman, "Forecasting americans' long-term adoption of connected and autonomous vehicle technologies," *Transportation Research Part A: Policy and Practice*, vol. 95, pp. 49–63, 2017. 1.1

[6] N. H. T. S. Administration, "Automated vehicles for safety," 2018, [Online; accessed 18- July- 2018]. [Online]. Available: https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety 1.2.1

[7] Tesla, "Tesla press information," 2018, [Online; accessed 18- July- 2018]. [Online]. Available: https://www.tesla.com/presskit/autopilot 1.2.1

[8] ——, “Full self-driving hardware on all cars,” 2018, [Online; accessed 18- July- 2018]. [Online]. Available: https://www.tesla.com/autopilot 1.2.1

[9] C. D. of Motor Vehicles, “Permit holders (testing with a driver),” 2018, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.dmv.ca.gov/portal/dmv/detail/vr/autonomous/permit 1.2.2

[10] ——, “Autonomous vehicles in california,” 2016, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.nmvb.ca.gov/roundtables/2016/industry/presentations/NMVB_Industry_Roundtable-Autonomous_Vehicles.pdf 1.2.2

[11] ——, “Adopted regulatory text,” 2018, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.dmv.ca.gov/portal/wcm/connect/a6ea01e0-072f-4f93-aa6c-e12b844443cc/DriverlessAV_Adopted_Regulatory_Text.pdf?MOD=AJPERES 1.2.2

[12] ——, “Autonomous vehicle disengagement reports 2017,” 2017, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.dmv.ca.gov/portal/dmv/detail/vr/autonomous/disengagement_report_2017 1.2.2

[13] Tesla, “Vehicle test data for 2017 reporting year,” 2017, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.dmv.ca.gov/portal/wcm/connect/f965670d-6c03-46a9-9109-0c187adebbf2/Tesla.pdf?MOD=AJPERES 1.2.2

[14] ——, “Customer privacy policy,” 2018, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.tesla.com/about/legal 1.2.2

[15] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2722–2730. 1.2.3

[16] N. Kalra and S. M. Paddock, "Research report: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" RAND Corporation, Tech. Rep., 2016. 1.2.4

[17] N. E. Boudette, "Waymo to buy up to 62,000 chrysler minivans for ride-hailing service," May 2018, [Online; accessed 20- July- 2018]. [Online]. Available: https://www.nytimes.com/2018/05/31/business/waymo-chrysler-minivans.html 1.2.4

[18] Tesla, "Master plan, part deux," 2016, [Online; accessed 19- July- 2018]. [Online]. Available: https://www.tesla.com/blog/master-plan-part-deux 1.2.4

[19] J. Rios-Torres and A. A. Malikopoulos, "Automated and cooperative vehicle merging at highway on-ramps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 780–789, 2017. 1.2.4

[20] M. Treiber and A. Kesting, *Traffic Flow Dynamics.* Springer, 2013. 2.1, 3.1.1, 3.1.2, 3.2.2, 3.2.3

[21] F. L. Hall, "Traffic stream characteristics," *Traffic Flow Theory. US Federal Highway Administration*, 1996. 2.2.1.3, 2.2.4

[22] L. Immers and S. Logghe, "Traffic flow theory," *Faculty of Engineering, Department of Civil Engineering, Section Traffic and Infrastructure, Kasteelpark Arenberg*, vol. 40, p. 21, 2002. 2.2.2

[23] B. Greenshields, W. Channing, H. Miller *et al.*, "A study of traffic capacity," in *Highway research board proceedings*, vol. 1935. National Research Council (USA), Highway Research Board, 1935. 2.2.4

[24] B. K. Horn, "Suppressing traffic flow instabilities," in *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on.* IEEE, 2013, pp. 13–20. 2.2.5, 2.2.5

[25] K.-S. Chang, W. Li, P. Devlin, A. Shaikhbahai, P. Varaiya, J. Hedrick, D. McMahon, V. Narendran, D. Swaroop, and J. Olds, "Experimentation with a vehicle platoon control system," in *Vehicle Navigation and Information Systems Conference, 1991*, vol. 2. IEEE, 1991, pp. 1117–1124. 2.3, 2.4

[26] P. Kavathekar and Y. Chen, "Vehicle platooning: A brief survey and categorization," in *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.* American Society of Mechanical Engineers, 2011, pp. 829–845. 2.3

[27] M. Zabat, N. Stabile, S. Farascaroli, and F. Browand, "The aerodynamic performance of platoons: A final report," 1995. 2.3

[28] "White paper: Automated driving and platooning issues and opportunities," ATA Technology and Maintenance Council Future Truck Program, Tech. Rep., 2015. 2.5

[29] W. Helly, "Simulation of bottlenecks in single-lane traffic flow," *Theory of TRAFFIC FLOW, PROCEEDINGS*, 1959. 2.5

[30] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1066–1077, 2017. 2.5

[31] R. E. Stern, S. Cui, M. L. D. Monache, R. Bhadani, M. Bunting, M. Churchill, N. Hamilton, H. Pohlmann, F. Wu, B. Piccoli *et al.*, "Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments," *arXiv preprint arXiv:1705.01693*, 2017. 2.5

[32] P. Kachroo, S. Agarwal, B. Piccoli, and K. Özbay, "Multiscale modeling and control architecture for v2x enabled traffic streams," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4616–4626, 2017. 2.5

[33] J. R. Taylor, *Classical Mechanics.* University Science Book, 2005. 3.1.1

[34] I. G. Jin and G. Orosz, "Optimal control of connected vehicle systems," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on.* IEEE, 2014, pp. 4107–4112. 3.2.1, 3.2.4, 5.1

[35] G. Orosz, R. E. Wilson, and G. Stépán, "Traffic jams: dynamics and control," 2010. 3.2.1

[36] D. Shiffman, *The Nature of Code: Simulating Natural Systems with Processings.* The Nature of Code, 2012. 3.2.3

[37] I. G. Jin and G. Orosz, "Dynamics of connected vehicle systems with delayed acceleration feedback," *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 46–64, 2014. 3.2.4, 3.2.4, 5

[38] K. Ogata, *The Nature of Code: Simulating Natural Systems with Processings.* PHI Learning Private Limited, 2014. 4.1

[39] H. J. Sussmann and J. C. Willems, "300 years of optimal control: from the brachystochrone to the maximum principle," *IEEE Control Systems*, vol. 17, no. 3, pp. 32–44, 1997. 4.1

[40] D. E. Kirk, *Optimal Control Theory.* Dover Publications Inc., 1998. 7

APPENDIX A

Software: Mathematical Modeling

A.1 Intelligent Driver Model (MATLAB)

```
% Simulation parameters
% SI units used throughout
dt = 0.1;   % time step
N = 2000;   % number of time steps
n = 3;      % number of cars
L= 70;      % radius of Ring (m)
d =  5;     % length of car (m)

% IDM parameters
v0 = 80; % desired speed
T = 2;   % reaction time
s0 = 15; % min. bumper-to-bumper distance
a = 0.2; % acceleration
b = 4.0; % comfortable braking

% Initial conditions
Vi = 0; % initial velocity
dx = [Vi, Vi, Vi];      % velocities
x = [0, 10, 55];        % positions
```

```
s(1) = x(2) - x(1) - d; % headways
s(2) = x(3) - x(2) - d;
s(3) = 2*pi*L - x(3) - d;

for j = 1:1:N % Time Loop
    for i  = 1:1:n % Loop for number of cars
        % Find the relative velocity
        if i == n  % if the car is lead car
            vl = dx(i) - dx(1);
        else
            vl = dx(i) - dx(i+1);
        end
        % Updates in this order:
        % desired gap
        % acceleration
        % velocity
        % position
        % headway
        s_star = s0 + max(0, dx(i)*T ...
               + 0.5*dx(i)*vl/sqrt(a*b));
        dvdt(i) = a*(1- (dx(i)/v0))^4 ...
                - a*(s_star/s(i))^2;
```

```
        dx(i) = dx(i) + dvdt(i)*dt;

        if dx(i) < 0
            x(i) = x(i) - 0.5*dx(i)*dx(i)/dvdt(i);
            dx(i) = 0;
        else
            x(i) = x(i) + dx(i)*dt + 0.5*dvdt(i)*dt*dt;
        end

        theta(i) = x(i)*2*pi/L; % x convert to angle
        % Reset to beginning of ring if necessary
        if theta(i) > 2*pi
           theta(i) = theta(i) - 2*pi;
        end
    end

    % Calculate the headways
    s(1) = x(2) - x(1);
    s(2) = x(3) - x(2);
    s(3) = x(1) - x(3);

    % Add the length of the road if the lead
```

```
    % vehicle is at the beginning of the
    % ring and the following vehicle is towards
    % the end
    if s(1) < 0
        s(1) = s(1) + L;
    end


    if s(2) < 0
        s(2) = s(2) + L;
    end


    if s(3) < 0
        s(3) = s(3) + L;
    end
    % Take care of states at each time step
    Theta(j,:) = theta(:);
    X(j,:) = x(:);        % Location
    V(j,:) = dx(:);       % Speed
    S(j,:) = s(:);        % Headways
    t(j) = dt*(j-1);      % actual running time
end
```

```
% Plot the velocities
figure('DefaultAxesFontSize',20)
plot_velocities = plot(t, V);
plot_velocities(1).LineWidth = 4;
plot_velocities(1).LineStyle = ':';
plot_velocities(2).LineWidth = 4;
plot_velocities(2).LineStyle = '--';
plot_velocities(3).LineWidth = 4;
plot_velocities(3).LineStyle = '-.';
xlabel('Time (s)')
ylabel('Speed (m/s)')
title('Speed of Vehicles')
legend('First Car','Second Car','Third Car',...
       'Location','northeast')
grid on

% Plot the headways
figure('DefaultAxesFontSize',20)
plot_headways = plot(t, S));
plot_headways(1).LineWidth = 4;
plot_headways(1).LineStyle = ':';
plot_headways(2).LineWidth = 4;
```

```
plot_headways(2).LineStyle = '--';
plot_headways(3).LineWidth = 4;
plot_headways(3).LineStyle = '-.';
title('Headway of vehicles')
xlabel('Time (s)')
ylabel('Headway (m)')
legend('First Car','Second Car','Third Car',...
       'Location','northeast')
grid on
```

A.2 Cellular Automaton Model (Python)

```
"""Definition of cellular automaton"""
import random
import time


class CellularAutomaton:
    def __init__(self, width, ruleset,
                 rand=(False, 0.5),
                 generation=0):
        self.generation = generation
        self.row = []
        self.width = width
        rand_init, density = rand
        if not rand_init:
```

```
            self.__create_row()
        else:
            self.__create_random_row(density)
        self.ruleset = ruleset


    def generate(self):
        next_gen = []
        left = self.row[-1]
        middle = self.row[0]
        right = self.row[1]
        state = self.__rules(
            left.state, middle.state, right.state)
        cell = Cell(state=state)
        next_gen.append(cell)
        for i in range(1, self.width-1):
            left = self.row[i-1]
            middle = self.row[i]
            right = self.row[i+1]
            state = self.__rules(
                left.state, middle.state, right.state)
            cell = Cell(state=state)
            next_gen.append(cell)
        left = self.row[-2]
        middle = self.row[-1]
        right = self.row[0]
        state = self.__rules(
            left.state, middle.state, right.state)
```

```
            cell = Cell(state=state)
            next_gen.append(cell)
        self.row = next_gen
        self.generation += 1

    def print_row(self):
        for cell in self.row:
            print(cell, end=' ')

    def terminal_run(self, iterations, delay=0.001):
        for i in range(iterations):
            self.print_row()
            time.sleep(delay)
            print()
            self.generate()

    def __create_random_row(self, density):
        for i in range(self.width):
            cell = Cell()
            cell.state = random.random() < density
            self.row.append(cell)

    def __create_row(self):
        middle = int(self.width/2) - 1
        for i in range(self.width):
            cell = Cell()
            if i == middle:
```

```
                cell.state = True
            self.row.append(cell)

    def __rules(self, left, middle, right):
        if left == True and middle == True and right == True:
            return self.ruleset[0]
        elif left == True and middle == True and right == False:
            return self.ruleset[1]
        elif left == True and middle == False and right == True:
            return self.ruleset[2]
        elif left == True and middle == False and right == False:
            return self.ruleset[3]
        elif left == False and middle == True and right == True:
            return self.ruleset[4]
        elif left == False and middle == True and right == False:
            return self.ruleset[5]
        elif left == False and middle == False and right == True:
            return self.ruleset[6]
        elif left == False and middle == False and right == False:
            return self.ruleset[7]

    def __str__(self):
        row = []
        for cell in self.row:
            row.append(cell.to_string())
        return ''.join(row)
```

```python
class Cell:

    def __init__(self, state=False, state_strs=('\u2588', ' ')):
        self.state = state
        self._on, self._off = state_strs

    def to_string(self):
        return self.__str__()

    def __bool__(self):
        return self.state

    def __repr__(self):
        return self.state

    def __str__(self):
        if self.state:
            return self._on
        else:
            return self._off
```

APPENDIX B

MATLAB Code: Control

B.1 PID Controller Simulation

```
% SI units used throughout

% Constants

% HDES places the vehicles evenly on the road

VLENGTH = 5; % m (vehicle length)

R = 70;

RLENGTH = R*2*pi; % m (road length)

VMAX = 5; % m/s (70 mph)

NUM_VEHICLES = 3;

HDES = (RLENGTH - NUM_VEHICLES*VLENGTH)/NUM_VEHICLES;


% System parameters

A = [

 0, 1, 0,  0, 0, -1;

 0, 0, 0,  0, 0,  0;

 0, 0, 0, -1, 0,  1;

 0, 0, 0,  0, 0,  0;

-1, 0, 1,  0, 0,  0;

 0, 0, 0,  0, 0,  0

];
```

```
B = [
0, 0, 0;
1, 0, 0;
0, 0, 0;
0, 1, 0;
0, 0, 0;
0, 0, 1
];

% Problem parameters
Q = 2.5e-5*eye(size(A));
Q(2, 2) = 1e-2; Q(4, 4) = 1e-2; Q(4, 4) = 1e-2;
H = Q;
R = 10*eye(3);
r = [HDES; VMAX; HDES; VMAX; HDES; VMAX];

% Initial positions
pos3 = RLENGTH;
pos2 = pos3 + 125 - RLENGTH;
pos1 = pos2 + 125;
```

```
h1_init = pos3 - pos1;
v1_init = 0;
h2_init = abs(pos2 - pos1);
v2_init = 0;
h3_init = abs(pos3 - pos2);
v3_init = 0;

X0 = [h1_init; v1_init; h2_init;
      v2_init; h3_init; v3_init];

% Boundary conditions
Kfinal = H;
Sfinal = -H*r;
tinitial = 0;
tfinal = 500;

% ODE options
rel_tol = 1e-6;
abs_tol = 1e-6*ones(1, 6);
abs_tol_k = 1e-6*ones(1, 36);
non_neg_idx = 1:6;
options1 = odeset('RelTol', rel_tol,...
```

```
                  'AbsTol', abs_tol, 'Refine',10);
optionsk = odeset('RelTol', rel_tol,...
                  'AbsTol', abs_tol_k, 'Refine',10);
% Apply constraint of non-negative numbers only
options2 = odeset(options1,'NonNegative',non_neg_idx);

% Symbols for solving system of differential equations
K = sym('K%d%d', [6, 6]);
S = sym('S%d', [6, 1]);

[tk, K] = ode45(@(t, K)kdot(t, K, A, B, Q, R),...
                [tfinal, tinitial], Kfinal, optionsk);
[ts, S] = ode45(@(t, S)sdot(t, S, A, B, K, Q, R, r, tk),...
                [tfinal, tinitial], Sfinal, options1);

[tx, X] = ode45(...
               @(t, X)xdot(t, X, K, S, R, A, B, tk, ts),...
               [tinitial, tfinal], X0, options2);
dt = 1e-1;
sim_time = tfinal*(dt^-1) + 1;
for j = 1:1:sim_time
    t_new=(j-1)*dt;
```

```
K_new = interp1(tk, K, t_new);
K_new = reshape(K_new, size(A));

S_new = interp1(ts, S, t_new);
S_new = S_new';

X_new = interp1(tx, X, t_new);
X_new = X_new';

u(:,j) = -(R^-1)*B'*K_new*X_new - (R^-1)*B'*S_new;

% Make sure that we don't go backwards!
if u(1,j) <= 0 && X_new(2,1) <= 0
    u(1,j) = 0;
end
if u(2,j) <= 0 && X_new(4,1) <= 0
    u(2,j) = 0;
end
if u(3,j) <= 0 && X_new(6,1) <= 0
    u(3,j) = 0;
end
tu(j)=t_new;
```

```
end

% Plot the headways
endtime = 250;
figure('DefaultAxesFontSize',16)
p1 = plot(tx, X(:, 1), tx, X(:, 3), tx, X(:, 5));
p1(1).LineWidth = 2;
p1(1).LineStyle = ':';
p1(2).LineWidth = 2;
p1(2).LineStyle = '--';
p1(3).LineWidth = 2;
p1(3).LineStyle = '-.';
title('Headways')
xlabel('Time (s)')
ylabel('Headways h_{i}(t) (m)')
legend('h_{1}^*(t)','h_{2}^*(t)','h_{3}^*(t)')
xlim([0, endtime])
grid on

% Plot the velocities
figure('DefaultAxesFontSize',16)
p2 = plot(tx, X(:, 2), tx, X(:, 4), tx, X(:, 6));
```

```
p2(1).LineWidth = 2;
p2(1).LineStyle = ':';
p2(2).LineWidth = 2;
p2(2).LineStyle = '--';
p2(3).LineWidth = 2;
p2(3).LineStyle = '-.';
title('Velocities')
xlabel('Time (s)')
ylabel('Velocities v_{i}(t) (m/s)')
legend('v_{1}^*(t)','v_{2}^*(t)','v_{3}^*(t)',...
       'Location', 'SouthEast')
xlim([0, endtime])
ylim([0, VMAX+1])
grid on

% Plot the control trajectories
figure('DefaultAxesFontSize',16)
p3 = plot(tu, u(1,:), tu, u(2,:), tu, u(3,:));
p3(1).LineWidth = 2;
p3(1).LineStyle = ':';
p3(2).LineWidth = 2;
p3(2).LineStyle = '--';
```

```
p3(3).LineWidth = 2;
p3(3).LineStyle = '-.';
title('Control trajectories')
xlabel('Time (s)')
ylabel('u^*(t) (m/s^2)')
legend('u_{1}^*(t)', 'u_{2}^*(t)', 'u_{3}^*(t)')
xlim([0, endtime])
print -depsc -r300 opt_ctrl_controls.eps
grid on

function dkdt = kdot(t, K, A, B, Q, R)
    K = reshape(K, size(A));
    dkdt = -K*A - A'*K - Q + K*B*(R^-1)*B'*K;
    dkdt = dkdt(:);
end

function dsdt = sdot(t, s, A, B, K, Q, R, r, tk)
    K = interp1(tk, K, t);
    K_new = reshape(K, size(A));
    dsdt = -(A' - K_new*B*(R^-1)*B')*s + Q*r;
    dsdt = dsdt(:);
end
```

```
function dxdt = xdot(t, X, K, S, R, A, B, tk, ts)
    K = interp1(tk, K, t);
    K = reshape(K, size(A));

    S = interp1(ts, S, t);
    S = S';

    U = -(R^-1)*B'*K*X - (R^-1)*B'*S;
    dxdt = A*X + B*U;
    dxdt = dxdt(:);
end
```

B.2 LQT Controller Simulation

```
% SI units used throughout
% Constants
% HDES places the vehicles evenly on the road
VLENGTH = 5; % m (vehicle length)
R = 70;
RLENGTH = R*2*pi; % m (road length)
VMAX = 5; % m/s (70 mph)
NUM_VEHICLES = 3;
```

```
HDES = (RLENGTH - NUM_VEHICLES*VLENGTH)/NUM_VEHICLES;

% System parameters
A = [
 0, 1, 0,  0, 0, -1;
 0, 0, 0,  0, 0,  0;
 0, 0, 0, -1, 0,  1;
 0, 0, 0,  0, 0,  0;
-1, 0, 1,  0, 0,  0;
 0, 0, 0,  0, 0,  0
];

B = [
0, 0, 0;
1, 0, 0;
0, 0, 0;
0, 1, 0;
0, 0, 0;
0, 0, 1
];

% Problem parameters
```

```
Q = 2.5e-5*eye(size(A));
Q(2, 2) = 1e-2; Q(4, 4) = 1e-2; Q(4, 4) = 1e-2;
H = Q;
R = 10*eye(3);
r = [HDES; VMAX; HDES; VMAX; HDES; VMAX];

% Initial positions
pos3 = RLENGTH;
pos2 = pos3 + 125 - RLENGTH;
pos1 = pos2 + 125;

h1_init = pos3 - pos1;
v1_init = 0;
h2_init = abs(pos2 - pos1);
v2_init = 0;
h3_init = abs(pos3 - pos2);
v3_init = 0;

X0 = [h1_init; v1_init; h2_init;
      v2_init; h3_init; v3_init];

% Boundary conditions
```

```
Kfinal = H;

Sfinal = -H*r;

tinitial = 0;

tfinal = 500;


% ODE options

rel_tol = 1e-6;

abs_tol = 1e-6*ones(1, 6);

abs_tol_k = 1e-6*ones(1, 36);

non_neg_idx = 1:6;

options1 = odeset('RelTol', rel_tol,...

                  'AbsTol', abs_tol, 'Refine',10);

optionsk = odeset('RelTol', rel_tol,...

                  'AbsTol', abs_tol_k, 'Refine',10);

% Apply constraint of non-negative numbers only

options2 = odeset(options1,'NonNegative',non_neg_idx);


% Symbols for solving system of differential equations

K = sym('K%d%d', [6, 6]);

S = sym('S%d', [6, 1]);


[tk, K] = ode45(@(t, K)kdot(t, K, A, B, Q, R),...
```

```
                [tfinal, tinitial], Kfinal, optionsk);
[ts, S] = ode45(@(t, S)sdot(t, S, A, B, K, Q, R, r, tk),...
                [tfinal, tinitial], Sfinal, options1);


[tx, X] = ode45(...
               @(t, X)xdot(t, X, K, S, R, A, B, tk, ts),...
               [tinitial, tfinal], X0, options2);
dt = 1e-1;
sim_time = tfinal*(dt^-1) + 1;
for j = 1:1:sim_time
    t_new=(j-1)*dt;
    K_new = interp1(tk, K, t_new);
    K_new = reshape(K_new, size(A));


    S_new = interp1(ts, S, t_new);
    S_new = S_new';


    X_new = interp1(tx, X, t_new);
    X_new = X_new';


    u(:,j) = -(R^-1)*B'*K_new*X_new - (R^-1)*B'*S_new;
```

```
    % Make sure that we don't go backwards!
    if u(1,j) <= 0 && X_new(2,1) <= 0
        u(1,j) = 0;
    end
    if u(2,j) <= 0 && X_new(4,1) <= 0
        u(2,j) = 0;
    end
    if u(3,j) <= 0 && X_new(6,1) <= 0
        u(3,j) = 0;
    end
    tu(j)=t_new;
end

% Plot the headways
endtime = 250;
figure('DefaultAxesFontSize',16)
p1 = plot(tx, X(:, 1), tx, X(:, 3), tx, X(:, 5));
p1(1).LineWidth = 2;
p1(1).LineStyle = ':';
p1(2).LineWidth = 2;
p1(2).LineStyle = '--';
p1(3).LineWidth = 2;
```

```
p1(3).LineStyle = '-.';
title('Headways')
xlabel('Time (s)')
ylabel('Headways h_{i}(t) (m)')
legend('h_{1}^*(t)','h_{2}^*(t)','h_{3}^*(t)')
xlim([0, endtime])
grid on

% Plot the velocities
figure('DefaultAxesFontSize',16)
p2 = plot(tx, X(:, 2), tx, X(:, 4), tx, X(:, 6));
p2(1).LineWidth = 2;
p2(1).LineStyle = ':';
p2(2).LineWidth = 2;
p2(2).LineStyle = '--';
p2(3).LineWidth = 2;
p2(3).LineStyle = '-.';
title('Velocities')
xlabel('Time (s)')
ylabel('Velocities v_{i}(t) (m/s)')
legend('v_{1}^*(t)','v_{2}^*(t)','v_{3}^*(t)',...
       'Location', 'SouthEast')
```

```
xlim([0, endtime])
ylim([0, VMAX+1])
grid on

% Plot the control trajectories
figure('DefaultAxesFontSize',16)
p3 = plot(tu, u(1,:), tu, u(2,:), tu, u(3,:));
p3(1).LineWidth = 2;
p3(1).LineStyle = ':';
p3(2).LineWidth = 2;
p3(2).LineStyle = '--';
p3(3).LineWidth = 2;
p3(3).LineStyle = '-.';
title('Control trajectories')
xlabel('Time (s)')
ylabel('u^*(t) (m/s^2)')
legend('u_{1}^*(t)', 'u_{2}^*(t)', 'u_{3}^*(t)')
xlim([0, endtime])
print -depsc -r300 opt_ctrl_controls.eps
grid on

function dkdt = kdot(t, K, A, B, Q, R)
```

```
    K = reshape(K, size(A));
    dkdt = -K*A - A'*K - Q + K*B*(R^-1)*B'*K;
    dkdt = dkdt(:);
end

function dsdt = sdot(t, s, A, B, K, Q, R, r, tk)
    K = interp1(tk, K, t);
    K_new = reshape(K, size(A));
    dsdt = -(A' - K_new*B*(R^-1)*B')*s + Q*r;
    dsdt = dsdt(:);
end

function dxdt = xdot(t, X, K, S, R, A, B, tk, ts)
    K = interp1(tk, K, t);
    K = reshape(K, size(A));

    S = interp1(ts, S, t);
    S = S';

    U = -(R^-1)*B'*K*X - (R^-1)*B'*S;
    dxdt = A*X + B*U;
    dxdt = dxdt(:);
```

```
end
```