EFFECTS OF COMMUNICATION DELAY AND KINEMATIC VARIATION IN VEHICLE PLATOONING

by

Megan R. Emmons

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Dr. Chris Winstead Major Professor	Dr. Kevin Heaslip Committee Member
Dr. Rose Hu Committee Member	Dr. Mark R. McLellan Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2013

Abstract

Effects of Communication Delay and Kinematic Variation in Vehicle Platooning

by

Megan R. Emmons, Master of Science

Utah State University, 2013

Major Professor: Dr. Chris Winstead
Department: Electrical and Computer Engineering

Vehicle platoons are efficient, closely-spaced groups of autonomously controlled vehicles which interact in a cooperative manner as they travel at high speeds down the road. Within this thesis, the robustness of a promising control algorithm for vehicle platooning is explored. The control algorithm was previously demonstrated in a controlled setting which significantly reduced the challenges facing full-scale implementation, most notably loss of shared data and imprecision within the data.

As found within this work, transmission loss and imprecise position, velocity, and acceleration data significantly degraded the control algorithm's performance. The overall platoon length contracted, inter-vehicle oscillations amplified, and relative energy expenditures increased by a factor of ten. Introducing a measure of each following vehicle's position with respect to the lead vehicle into the control algorithm noticeably reduced platoon contraction. Utilizing an adaptive gain and performing some signal processing on the data further improved the platoon's performance. Combining these modifications with a model of the proposed communication scheme shows platoons of up to 25 vehicles are feasible.

(90 pages)

Public Abstract

Effects of Communication Delay and Kinematic Variation in Vehicle Platooning

by

Megan R. Emmons, Master of Science

Utah State University, 2013

Major Professor: Dr. Chris Winstead
Department: Electrical and Computer Engineering

Vehicle platoons are efficient, closely-spaced groups of robotically controlled vehicles which travel at high speeds down the road, similar to carts in a train. Within this thesis, a promising control algorithm for vehicle platooning is explored. The control algorithm was previously demonstrated in a sterile setting which significantly reduced the challenges facing full-scale implementation of platoons, most notably loss of shared data and imprecision within the data.

As found within this work, transmission loss and imprecise position, velocity, and acceleration data significantly degraded the control algorithm's performance. Vehicles in the platoon became more closely spaced, changed speeds more frequently, and expended far more energy than necessary. Introducing a measure of each following vehicle's position with respect to the lead vehicle into the control algorithm noticeably reduced platoon contraction. Adjusting the control algorithm's responsiveness based on what data was successfully received reduced the speed-variations by vehicles. Finally, using past behavior to predict the next acceleration reduced the energy used by each vehicle. Combining these modifications with a model of the proposed communication scheme shows platoons of up to 25 vehicles are feasible.

This work is dedicated first and foremost to my father.
As a testament to the decisions I have made in life,
this thesis is also dedicated to the little voice inside that prevailed,
despite those moments of doubt.

Acknowledgments

A work of this size is not possible without the help and support of many wonderful individuals. A big debt of gratitude is owed to all my teachers and professors who fanned my spark of curiosity. Most recently at Utah State University, I would like to thank Dr. Cripps, Dr. Chen, and Dr. Moon for confirming my passion for robotics and providing me with the tools and confidence to continue pursuing the evolving field.

I especially want to thank my advisor, Dr. Winstead, for generously bringing me into the AET research group. Although not his specialty, he encouraged me to bring my controls interest to bear on the project. I appreciate his patience and guidance on this work. I would also like to thank committee members, Dr. Heaslip and Dr. Hu, for their assistance, support, and time as well as fellow researchers within the AET group. Mary Lee was an invaluable ally, and I sincerely appreciate everything she did to help me through this process!

Outside the academic community, I certainly could not have completed this work without the incredible amount of patience and support from my friends and family! Most especially, I owe a big part of this work to my father who truly was my rock. Even when I felt completely lost or confused, he always knew exactly what to say to help me push forward. Despite the miles between us, conversations with my dear friend, Mallory, were equally invaluable. From random programming questions, deep philosophical talks about life, to light chit-chat, Mallory kept me motivated and grounded.

Finally, the climbing community in Logan has truly become like family. Ros, April, Greta, Leanna, Trevor, Mike, Brad, Todd, and Tony all helped turn Logan into a home. They helped me maintain a better balance between life and work. When my mind ran in circles, hanging out with any one of these dear friends brought everything back into focus; so thank you all!

Megan 'Marmot' Emmons

Contents

List of Tables

List of Figures

Notation

Events

TP	throughput
S	average number of successful messages
T	communication and control period
Δv	difference in velocity
a	general acceleration
t	current time
Δs	difference in position
$\ddot{x}^*$	vehicle's desired acceleration
$\ddot{x}$	vehicle acceleration
i	vehicle number
N	platoon size
C_1	control gain
ξ	damping ratio
ω_n	controller bandwidth
$\dot{\varepsilon}$	spacing error
L	ideal vehicle spacing
γ	sinusoidal frequency modifier
δ	time step for numeric integration
T_r	reaction time
R	remaining time
E	relative energy
σ	magnitude of standard deviation
K	tunable gain for modified controller

Acronyms

CSMA/CA	carrier sense multiple access with collision avoidance
CW	current backoff window
DIFS	distributed inter-frame space
DSRC	dedicated short-range communication
DSSS	direct sequence spread spectrum
ERP	extended rate PHY
FHSS	frequency hopping spread spectrum
ITS	intelligent transportation systems
IVC	inter-vehicle communication
MAC	media access control
OFDM	orthogonoal frequency division multiplexing
OSI	open systems interconnect
PATH	Partners for Advanced Transportation Technology
PHY	physical layer
PLCP	physical layer convergence protocol
PPDU	PLCP Protocol Data Unit
PVA	position, velocity, and acceleration
SARTRE	Safe Road Trains for the Environment
WAVE	Wireless Access in Vehicular Environments
WLANs	wireless land area networks

Chapter 1

Introduction

1.1 Project Overview

Distributed multi-agent systems are a challenging, yet promising, class of control systems engineering. While classic control is limited to single-input, single-output linearized systems, distributed control is a subset of modern control theory and so accommodates more complicated systems with multiple inputs, multiple outputs, and a variety of potential nonlinearities. Each member of a multi-agent network is its own system containing sensors, a control algorithm, and a means of actuation. This swarm approach is more robust because there is no central controller but presents its own challenges because each agent's control algorithm requires information gathered from other members. In this sense, the actions of each agent must be made using local information, but the actions themselves have a global impact. The dependence on communicated information from other members introduces the concept of time-sensitive data and starts placing bounds on swarm size, communication protocol, and dispersion area. While multi-agent systems have many different classifications and applications, this work focuses on vehicle platooning.

The general goal of vehicle platooning is to form groups of efficient, closely spaced, autonomous vehicles which interact in a cooperative manner. The feasibility of this concept was demonstrated in 1997 when California Partners for Advanced Transportation Technology (PATH) demonstrated an eight-vehicle platoon [1]. Although PATH achieved its goals, the demonstration site was a controlled environment. Their sterile test track eliminated many of the complications facing full-scale implementation of such a system, most notably obstacle avoidance and navigation. Both these challenges can be mitigated through the development of a robust control algorithm.

While stability is a necessary criteria for nearly every control system, guaranteeing

stability of the individual vehicles' control algorithms is not sufficient to ensure the platoon itself is stable. External disturbances, system nonlinearities, and imprecision in sensor measurements all contribute to small deviations between actual physical properties and calculated properties. This difference is considered error. When evaluating stability, it is important to ensure the system's error is driven to zero as time approaches infinity. The stability of each vehicle's control algorithm should adhere to this rule but, because the system consists of multiple vehicles, additional requirements are necessary to ensure stability of the entire platoon. For this reason, string stability is instead used when evaluating proposed control algorithms because it ensures errors decrease as information is propagated down the platoon, thus ensuring the entire platoon is stable.

Control algorithms have been the primary focus for ensuring string stability within vehicle platoons but, as Rajamani et al. concluded, string stability can only be achieved with the inclusion of inter-vehicle communication (IVC) [2]. In other words, platoon vehicles need kinematic information from other vehicles within the platoon. Ensuring fast, reliable communication will also aid in obstacle avoidance and navigation. Communication presents additional difficulties, however, because some delay is always present when sending messages. Part of the delay is due to the physical transmission time and processing of data. More notably, there is the time needed for a sender to gain access to the designated channel if the communication protocol involves a random access medium and for the validation of received data if security measures are implemented. In total, the communication delay could potentially compromise the string stability of a platoon. X. Liu et al. noted that the presence of a fixed delay significantly compromised string stability [3]. Wireless networks, which have random delays and the added challenge of packet losses, will exacerbate the situation.

All distributed multi-agent systems require communicated information but some systems are more tolerant of the accompanying delays. In many exploration-based applications, agents operate in pseudo-discrete-time by simply not acting until they receive complete information. This is not the case in vehicle platooning. Vehicles are moving at high speeds

and must act in continuous time. For example, a platoon of vehicles traveling at 60 mph covers more than 26.5 meters in just one second. According to an evaluation of the proposed communication protocol for vehicle platooning, the delay of control messages of highest priority remains on the order of tens of milliseconds [4]. In this time frame, vehicles will have traveled 0.265 meters, a significant displacement when the spacing between vehicles is envisioned to be on the order of 0.1 meters for braking concerns [5]. Each vehicles control algorithm must therefore be robust enough to compensate for these potential delays yet there are currently no simulations which combine proposed control algorithms with the anticipated communication protocol.

This work seeks to analyze challenges facing full-scale implementation of vehicle platooning. Specifically, a flexible simulator is developed so the influence of imprecise kinematic data on the control algorithm as well as the interaction between control and communication can be explored for vehicle platooning applications. The rest of this chapter presents some general background regarding the evolution of vehicle platooning and relevant terminology. Chapter 2 discusses the communication model in detail and presents key results regarding successful transmission rates as a function of platoon size and transmission period. The original PATH control algorithm is presented in Chapter 3. Results from the final vehicle platooning simulator, which combines the control and communication models, are the focus of Chapter 4. Chapter 4 also presents proposed modifications to the control algorithm for improved platoon performance in the presence of transmission loss and imprecise data. Finally, conclusions and necessary future work are presented in Chapter 5.

1.2 Background

From a control engineering perspective, vehicle platooning is a unique subset of distributed multi-agent systems. In platooning, vehicles travel along an established path and exhibit comparatively predictable behavior. These features make platooning simpler than many other multi-agent systems currently being explored because fewer unknown scenarios exist but platooning has its own set of challenges: events, such as an emergency braking scenario, occur at very high-speeds and the consequences of an error are much greater. As

suggested in the introduction, many multi-agent systems operate in a discrete domain but high vehicle velocities require vehicles to be continually responsive to their surroundings, hence platoon vehicles must operate in continuous time. This innate property makes communication delay a serious threat to system safety because vehicles must decide on a course of action with or without up to date information. Furthermore, like all multi-agent systems, vehicles are free to make the wrong decision. While the wrong decision may result in the loss of a single agent in most systems, in vehicle platooning the wrong decision will likely result in a mass collision of the entire platoon. Despite these challenges, much effort is being undertaken to find solutions which will lead to a fully implemented platooning system due to its many potential benefits.

One of the most direct benefits of vehicle platooning is the alleviation of traffic congestion. Over 2.9 billion gallons of fuel were wasted in congested traffic across the nation in 2012, corresponding to a total financial cost of $121 billion [6]. Forming groups of autonomous vehicles allows for closer vehicle spacing, thus increasing roadway capacity so more vehicles can utilize existing infrastructure, and decreasing the amount of speed variations, and hence the occurrence of stop-and-go congestion. Platooning also improves vehicle efficiency by placing cars into a drafting scenario where each following vehicle benefits from decreased drag. The reduced speed variation from autonomous control further improves vehicle efficiency while increasing passenger comfort and lowering the number of traffic collisions [1, 7].

PATH undertook many of the preliminary investigations into vehicle platooning within the United States during the early 1990s. This group laid significant groundwork in the characterization of a hierarchical architecture for automated highway systems, established terminology, quantified key variables for vehicle modeling, and identified important areas for research [8]. Early publications from this group even laid out basic guidelines for necessary communication [9]. Europe's SARTRE (Safe Road Trains for the Environment) group began with similar goals as PATH, but their work culminated in a more realistic demonstration.

Through SARTREs three-year endeavor, four Volvos successfully followed a professionally driven truck on public roads in Spain [7]. Using wireless communication and an autonomous controller developed by Ricardo, a global engineering consultant firm, the cars were able to successfully follow the lead vehicle, sometimes with a gap of no more than four meters, while traveling at speeds up to 90 km/h in the presence of other roadway traffic. Although the platoon successfully navigated over 200 km in a single day, little documentation of the control algorithm employed or the specific scenarios tested has been published. It does not appear like the platoon formation and lane changing were addressed autonomously, however. The impact of emergency braking, which will be discussed further in Chapter 3, also raises questions regarding the ideal vehicle spacing and hence the speed at which the controller must operate [5]. As the number of platoons increase, there will also be significantly more traffic on the communication channel. This increased load can drastically increase the time required to successfully transmit a message, and hence destabilize the control algorithms.

To operate, each vehicle within the platoon must have information regarding the anticipated actions of other platoon members. When that information is not successfully received by a vehicle, the car must still make a decision regarding its next actions. For many situations, prediction algorithms can be used to form a reasonable approximation of the missing information. A successful prediction will not account for an emergency which by its nature implies a drastic change in the situation. If the lead vehicle suddenly brakes and cannot transmit that action, for example, the following vehicles cannot modify their actions and so collide. A communication protocol which minimizes the occurrence of dropped transmissions is therefore necessary before vehicle platooning can be safely implemented on a large scale. One of the leading candidates for communication protocols in the United States is an amendment to the IEEE 802.11 family of standards. This amendment, known as 802.11p or WAVE (Wireless Access in Vehicular Environments), is designed specifically for vehicular communication and was approved in July 2010 [10]. Established wireless networks are already governed by IEEE 802.11 and have demonstrated an ability to accommodate a

large number of users. Building on this platform, WAVE simply modifies the bandwidth allotments of IEEE 802.11a without altering the protocol which has been so successfully implemented. While several research papers have proposed the addition of more layers to WAVE, the research community has been very receptive to the protocol as a foundation for vehicle platooning. Further details on the IEEE 802.11p protocol can be found in Chapter 2 as it was the chosen protocol for the platooning simulation presented herein.

No protocol exists which will guarantee instantaneous communication. Although IEEE 802.11 is designed to handle a large number of users, sizable delays do occur. In the case of platoon vehicles, the delay may prevent necessary information from being received by the time it is needed to modify the control algorithm. Therefore, a control algorithm must be developed which can tolerate some loss of information. One of the foundational researchers in the control realm, Rajesh Rajamani, has published an entire textbook dedicated to vehicle control [11], and helped develop the control algorithm used in the PATH demonstration. Each vehicle in the PATH demonstration contained a lateral and a longitudinal control to minimize path deviation and maintain vehicle spacing, respectively. The lateral control can be addressed locally but the longitudinal controller relies on information from other platoon vehicles and returns a desired acceleration which the vehicle then tries to match. Within Rajamani's control algorithm, it is assumed the lead vehicle knows the exact trajectory information.

Rajamani's control algorithm was successful for the PATH demonstrations and continues to attract current researchers. P. Fernandes and U. Nunes implemented the same longitudinal control in simulations where they proposed strategies for mitigating communication delays [12]. Although promising in simulation, the main theme of Fernandes and Nunes' work was the introduction of a standard 100 ms delay. As was previously mentioned, this long of a delay can have disastrous consequences in an emergency situation. Nonetheless, their work reveals the robust nature of Rajamani's controller. That robustness, combined with the physical demonstration, is why the same longitudinal control design was implemented in this work and will be discussed in more detail in Chapter 3.

Chapter 2

Communication within Vehicle Platoons

While platooning, vehicles will travel through a wide-range of terrain, from open rural roads to densely packed urban streets with a corresponding variation in speed. Communicating in this changing environment requires a flexible, high-speed protocol much like the ad hoc structure of Wireless Land Area Networks (WLANs). An ad hoc network offers the additional benefit of not requiring large amounts of supplementary infrastructure so vehicle platooning can be gradually incorporated into existing roadways as planned. Current supporting technology is already available and can successfully accommodate a large number of users. When vehicular communication emerged as a topic of interest, these factors led many researchers to recommend the modification of existing wireless protocols for use in vehicular networks.

WLANS are governed by the IEEE 802.11 family of standards. These standards allot frequency bands, stipulate channel access protocols, define accepted packet structures, and specify data rates. Most importantly, 802.11 provides a strong foundation of standards while permitting evolution to incorporate promising new technologies. As mentioned in Chapter 1, IEEE recently introduced an amendment to the 802.11 standard, known as IEEE 802.11p, specifically for applications in vehicular networks.

The goal of this thesis is to model the interactions between communication and control in vehicle platoons. Many communication experts have developed detailed models for the IEEE 802.11 family while others have proposed additional layers for the newly introduced IEEE 802.11p protocol [13–18]. This research is important and necessary for vehicle platooning but such detailed analysis obscures how other subsystems, such as the control algorithms within each vehicle, will be affected by the nature of the communication protocol. To this end, the model herein focuses only on the physical layer (PHY) and Media Access

Control (MAC) sublayer of IEEE 802.11p as these two layers are primarily responsible for data transmissions. The PHY dictates the amount of information added before and after a transmission for the data to be successfully extracted by the receiver. Meanwhile, the MAC governs the scheduling of transmissions. These two layers are hence communication factors which have a large impact on the control algorithms in vehicle platooning.

An overview of the established IEEE 802.11a standard will be presented in this chapter because it serves as the foundation for 802.11p. A general understanding of the layers, transmission times, and packet sizes for the established standard helps clarify the 802.11p amendment. The communication model, though targeted toward 802.11p, is flexible enough to provide a loose model of 802.11a which has more documentation so some model verification can be gleaned by comparing throughput results from the simulation to published literature. In Section 2.2, a brief history and detailed description of the 802.11p amendment is presented before providing an in depth explanation of the communication simulation. The chapter concludes with an analysis of the two protocols as implemented in C++ code for use with the overall vehicle platooning model.

It should be noted that platooning will most likely require a combination of vehicle to vehicle communication as well as vehicle to infrastructure communication. This work focuses on vehicle to vehicle communication and assumes the lead vehicle will receive exact information from the infrastructure regarding desired speed, upcoming vehicle maneuvers, and potential emergency situations. The platoon is further assumed to act as an isolated ad hoc network with data being broadcast from one vehicle to the remaining platoon members. No acknowledgement packets are therefore considered. Instead, if a vehicle successfully gains access to the communication channel, it is assumed the vehicle can successfully transmit the desired message to all vehicles and all vehicles receive that message at the same time.

2.1 Discussion of the IEEE 802.11a Standard

IEEE 802.11a uses a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol wherein the Physical layer Convergence Protocol (PLCP) sublayer continually polls the communication medium to determine whether the channel is busy or idle. The PLCP

reports the channel status to the Media Access Control (MAC) sublayer which then decides whether to attempt a transmission. The channel must be sensed idle for a distributed inter-frame space (DIFS) of 34 μs before the MAC will begin the transmission procedure [19].

When the channel has sensed an idle DIFS, the transmitter uniformly chooses a backoff time in the interval (0,CW-1) where CW is the current backoff window size. Initially, CW is equal to the minimum value of 15. Should the transmission be unsuccessful, CW is doubled until a maximum window size is reached or the transmission is successful. The problem of CW's units has not been resolved so the resulting value is scaled to be on the same order as the other message parts, specifically microseconds. Once the transmitter has waited for its determined offset period, the transmission begins. First, the preamble and PHY header are transmitted at the lowest supported data rate of 6 Mbps, thus communicating at a rate for which all receivers are listening. These two packets contain information about the speed at which data will be sent, the encryption used on the data, and other message-critical information so the receiver can successfully extract the data from received communication packets.

The 802.11a standard does not describe a specific PHY but instead leaves this as a design option. The majority of physical layers use Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS), or Orthogonal Frequency Division Multiplexing (OFDM) [20]. FSSS has some security advantages because the communicated data changes frequencies; however, coordinating the frequency hopping requires larger headers and longer synchronization, lowering the overall throughput of the system making large networks less efficient. If a DSSS Physical layer is chosen, 802.11 Clause 15 allows for either Differential Binary Phase-Shift Keying or Differential Quaternary Phase-Shift Keying modulation which correspond to 1 or 2 Mbps data rates respectively [21]. By contrast, an OFDM Physical layer provides up to 54 Mbps data rates and allows for multiple subcarriers making it more suitable for vehicular platooning. Most importantly for this work, the PHY specified in IEEE 802.11p is OFDM.

Two OFDM Physical layers are available in the IEEE 802.11 standard, both providing

up to 54 Mbps data rates, but only the Extended Rate PHY (ERP) OFDM Physical layer is used in 802.11a. As shown in Figure 2.1, the ERP-OFDM PPDU is comprised of a preamble, header, and data field. Similar to the initial preamble, the PLCP preamble provides a known, repeating sequence so the receiver can acquire the incoming OFDM signal and synchronize its demodulator. This training sequence may take up to 16 μs to complete [21].

Following the preamble, the PLCP begins transmitting the ERP-OFDM header. Figure 2.2 shows the 24 bits which make up this field. The first 4 bits of the header indicate the modulation and coding rate of the remaining PPDU. After the rate subfield is a reserved bit which is set to zero as it is not currently used. Next is the 12 bit length subfield which indicates the number of octets in the actual data message. The parity bit is an even parity bit based on the preceding 17 bits. Finally, the header sends a string of 6 zeros which comprise the signal tail subfield. After sending the header, the transmitter switches to the specified data rate and begins sending the data. The data field itself consists of a service subfield, the actual data, a tail subfield, and then a series of pad bits. The service field is currently a string of 16 zeros, the first seven bits are used to synchronize with the receiver while the remaining nine are currently unused, and is transmitted at the data rate specified in the ERP-OFDM PPDU header. The data is also sent at this rate and cannot exceed 4095 octets [21]. Finally, the data tail sends a string of 6 bits, all 0, to reset the encoder then fills in the pad bits. The pad bits subfield is at least six bits long but is the minimum number of bits needed so the data field is a multiple of the number of bits in a coded OFDM symbol, namely 48, 96, 192, or 288 [21]. After the data field has been sent, the PLCP sends a confirmation primitive to the MAC layer and relinquishes control of the channel.

2.2 Wireless Access in Vehicular Environments

Extending the IEEE 802.11 standard into vehicular environments via IEEE 802.11p was a logical step. It builds on an established, proven foundation and so benefits from wireless expertise and guarantees vehicles from different manufacturers can interact with each other and with future infrastructure units. Most importantly, it ensures vehicular

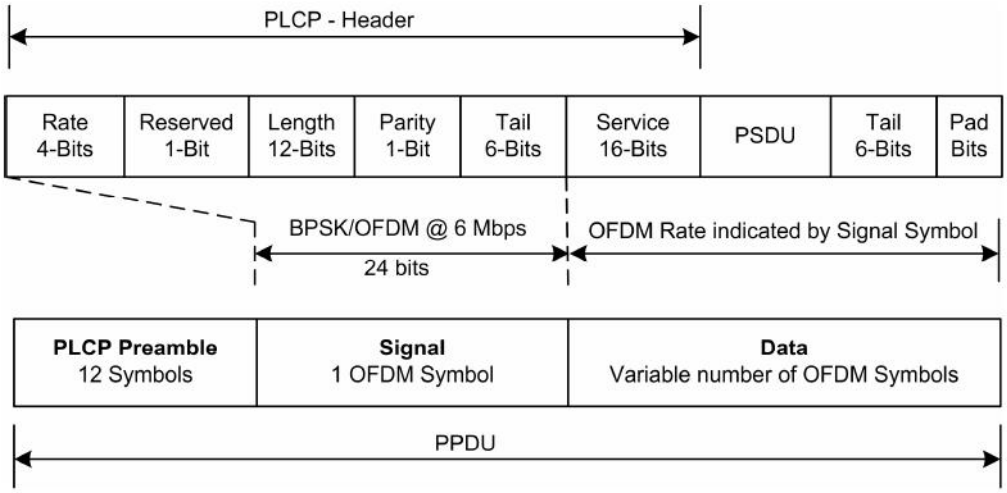


Fig. 2.1: ERP-OFDM PPDU as specified in the 802.11a standard.

Fig. 2.2: ERP-OFDM PPDU header for 802.11a.

networks will benefit from ongoing developments in the wireless family. Despite these advantages, IEEE 802.11 in its original form was unable to handle the longer range of operation, high vehicle speeds, multipath interference, overlapping networks, and the general nature of the applications [22] encountered in vehicular environments.

Task group p of the IEEE 802.11 working group began developing an amendment to the 802.11 standard specifically for vehicular environments. The amendment built on the Federal Communications Commission's allocation of 75 MHz of bandwidth in the 5.9 GHz band for dedicated short-range communications (DSRC) in October 1999. The allocation was part of the National Intelligent Transportation Systems Architecture (NITSA) developed by the Intelligent Transportation Society of America, the U.S. Department of Transportation, and other parties tasked with creating an Intelligent Vehicle Highway System, now known as Intelligent Transportation Systems (ITS) [22]. Specifications for additional necessary layers were developed by the 1609 working group, also associated with IEEE. Collectively, IEEE 802.11p and IEEE 1609 provide standards for wireless access in vehicular environments herein referred to as WAVE.

WAVE supports two stack architectures, namely the standard Internet Protocol version six and the proprietary WAVE Short-Message-Protocol. Accommodating two stack structures allows WAVE to provide high-priority, time-sensitive communications along with less demanding, service-oriented transmissions [22]. Both stacks rely on the same first two Open Systems Interconnection layers, however. That is, the same physical layer and MAC are used for all WAVE applications - another reason for focusing this model on the PHY and MAC sublayer. The upper OSI layers are predominantly contained within the 1609 family while IEEE 802.11p defines the PHY and MAC sublayer for WAVE.

Building on the protocol of 802.11a, the IEEE 802.11p amendment still utilizes a CSMA/CA MAC but specifies an OFDM physical layer [23]. Only one physical layer is allowed in WAVE to ensure uniformity in communication, thus simplifying necessary equipment on vehicles. OFDM was chosen for the PHY because it has several benefits for vehicular networking applications: high spectral efficiency because subchannels can overlap,

resiliency to RF interference, and lower multipath distortion [21]. A total of 64 subcarrier channels are generated by the OFDM modulation as described in the standard. Of these 64 channels, eleven are guard subcarriers to buffer the edges of the channel, a zero channel is set aside in the middle of the range to serve as a DC subcarrier, and the remaining 52 subchannels are used for data transmission.

Channel assignment is addressed in higher layers of the OSI. For the model presented in this work, the primary modification to the PHY and MAC which occurs from the established 802.11a standard to the 802.11p amendment is in the channel bandwidth and data rates. While 802.11a utilized 20 MHz channels, 802.11p uses 10 MHz channels and half the data rates provided in 802.11a [22,24]. Therefore, the preamble and header times described in the previous section are carried into the 802.11p protocol. The only value significantly affected by the amendment is the maximum data rate being reduced to 27 Mbps.

2.3 Implementation of Protocols in Code

When looking at the challenges of vehicle platooning from a control algorithm perspective, the primary role of communication is to share accurate kinematic data between vehicles in a timely manner. Intuitively, increasing the number of vehicles within a platoon will also increase the communication delay but the exact relationship is not known. It is therefore advantageous to explore the impact of increasing platoon size on communication delay and transmission failure rate - data that can then be incorporated into a more complete simulation to investigate the robustness of proposed control algorithms. In that vein, the first piece of the simulation was the communication protocol itself and focused on allowing an increasing number of vehicles to attempt transmissions following the WAVE protocol.

As summarized in the previous section, IEEE 802.11p governs the physical layer and MAC sublayer within WAVE and hence is responsible for the basic transmission of information. Further, 802.11p builds on the general 802.11 protocol which uses a carrier-sense, multiple access with collision avoidance MAC to schedule transmissions and access the communication channel. Although 802.11p specifies an OFDM PHY, from a data transmission standpoint this only affects the structure of the preamble and header which precede every

data transmission. Specifically, the Physical Layer Convergence Protocol, which is responsible for synchronizing the demodulator to the incoming OFDM signal, requires a 12 symbol preamble. Matching the transmitter and receiver in this manner may take up to 16 μs [21]. The PLCP header occupies 24 bits as previously described. These two components are fixed requirements of the communication protocol and are independent of the data being transmitted.

The exact nature of data needed by vehicles is still a subject of much research because it, in large part, depends on the control algorithm chosen and the security measures implemented. Nonetheless, a reasonable structure can be determined for the sake of simulation. For this work, it is assumed vehicles will broadcast information about their position, velocity, and acceleration (PVA). Every control algorithm investigated for this work can operate using some combination of this information. Reserving 16 bits for each piece of information comfortably allows for one part in 2^{16} of precision. Each message is also assumed to contain a status update for the vehicle, vehicle identification number, and message identification number. The vehicle status is currently unused but can serve as an indicator of whether a vehicle is entering or leaving the platoon, alert other cars if the vehicle encounters a mechanical problem which does not immediately impact the integrity of the platoon, or provide additional services. For the sake of uniformity and to simplify hardware design, these components are assumed to be 16 bits as well. Combined, a total of 96 bits are therefore reserved for the actual data.

Following the packet structure dictated by 802.11a, and therefore by 802.11p, 6 bits are appended to the end of the data making the total data field 96 bits in length. An additional 90 bits are needed as pad bits to ensure the total data packet is a multiple of an OFDM symbol, in this case 192 bits. The overall packet structure is shown in Figure 2.3. The time required to send the 216 bits of data is a function of the data rate indicated in the header. Recall 802.11p offers a range of data rates from 3 Mbps up to 27 Mbps, half those available in 802.11a, corresponding to a range in transmission times from 72 μs down to approximately 8 μs, also shown in Figure 2.3.

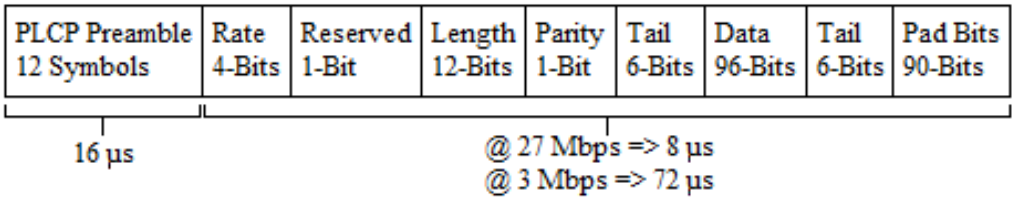


Fig. 2.3: Packet structure for 802.11p model.

Here it is worth emphasizing a few points regarding the data assumptions made within the simulation. The number of bits contained in a data packet must be a multiple of the bits in an OFDM symbol, namely 48, 96, or 192 bits. Given the need to communicate some amount of kinematic information and the appending of at least 12 bits to the end of the data, a 48-bit message contains insufficient information. Only 36 bits are available so PVA data would be restricted to 8 bits, leaving at most 12 bits for additional information. This drastic reduction in data only lowers the entire packet delivery time from 88 μs to approximately 40 μs at 3 Mbps. A more reasonable data packet would be 96 bits. PVA data can return to a 16-bit representation while still allowing up to 36 bits for sending additional information. This is reasonable but provides very little improvement in transmission times: at 3 Mbps the difference between a 192-bit packet and a 96-bit packet is 32 μs, less than 4 μs at the maximum data rate of 27 Mbps. This scenario of diminishing returns on data rates is discussed in detail by Y. Xiao and J. Rosdahl [19]. Further, the purpose of the simulations is to explore the robustness of control algorithms and so the worst-case scenario is presented, namely the largest data packet and hence the largest delay. For these reasons, the data packet of Figure 2.3 is used throughout the simulations for timing purposes.

For the simulation, the purpose of the packet structure of Figure 2.3 is to obtain a total message time which is the combination of the fixed 16 μs preamble as well as the time required to send the 192-bit data packet. Each car within the platoon attempts to send one message every communication period. The communication period is directly connected to the fact that each vehicle must decide on a course of action for the next time interval and hence, must receive PVA updates from the relevant vehicles prior to this decision. If a vehicle fails to transmit in the designated communication period, the intended message is

aborted because the information is now stale. Instead, a new update is scheduled for the next communication period.

At the beginning of the program, each vehicle in the platoon randomly selects a backoff time in the interval (0, CW-1) where CW is the current contention window size. CW is initially set to 15 because no transmissions have yet been rescheduled. If a vehicle fails to gain control of the channel at the desired time, the transmission is rescheduled using the same procedure but CW is doubled. CW continues to double with each rescheduling until a maximum CW of 1023 is reached. This timing is based on IEEE 802.11a [19] but is not altered in the 802.11p amendment [22]. The uniformly distributed offset times are all scaled to guarantee the transmission will be attempted at least once in the desired communication period. Each vehicle will attempt to transmit their message at their offset time beyond the current time.

Once every vehicle has determined an initial offset time, an event queue is constructed which contains vehicle identity, event type, transmission attempt, event time, and time for the initial attempt. A sample event queue for a three car platoon is shown in Table 2.1. The vehicle identity is simply the vehicle's location in the platoon with "0" designating the lead vehicle. Event type is reserved for future explorations but in the current simulation is left as a "1" to indicate a communication event is scheduled. Initially, the transmission attempt for every vehicle is set to "0." This value is incremented each time a transmission is rescheduled so the appropriate contention window can be used in the rescheduling process. The random offset time is stored in both the event time and initial attempt columns. Initially these two columns are identical because no rescheduling has been necessary, but if a rescheduling occurs, the new transmission time will be stored in the event time column while the initial attempt column will remain unaltered. The initial attempt column serves as a valuable

Table 2.1: Sample event queue for three car platoon.

vehicle_number	event_type	transmission_attempt	event_time	initial_attempt
1	comm.	0	0.000747554	0.000747554
0	comm.	0	0.000748155	0.000748155
2	comm.	0	0.00079928	0.00079928

analytical tool; it shows the delay between when a vehicle wants to transmit and when the transmission is successful.

The rows within the event queue are then arranged chronologically so the first row contains the earliest transmission. In the case of Table 2.1, the sorting has already occurred. As the first vehicle in the event queue, Car 1 is clearly the first vehicle attempting a transmission. Car 1 will monitor the channel prior to its transmission. If no transmission has occurred within DIFS, defined as 34 μs, of the desired transmission time, Car 1 successfully sends its message. Because no prior transmissions are scheduled in the scenario presented in Table 2.1, the channel is sufficiently clear so Car 1 does transmit successfully. When a vehicle successfully gains control of the communication channel, the channel is then occupied for the time needed to send the data message. For reasons described previously, every vehicle's message requires 88 μs to transmit. If the next transmission is scheduled during this 88 μs or 34 μs after, that transmission must be rescheduled.

When a message is rescheduled, the transmission attempt is incremented to ensure the contention window is doubled. A new random offset is then chosen in the range (0, CW) where CW has been doubled. The offset is scaled as previously then added to the current time to determine when the next transmission attempt will occur. There is no guarantee a transmission will be rescheduled into the same communication period. In the three car platoon example shown in Table 2.1, Car 1 successfully gained control of the communication channel 747.554 μs into the simulation and begins transmitting. The channel is therefore occupied from 747.554 μs to 835.554 μs which clearly interferes with both Car 0 and Car 2. When Car 0 begins preparing to transmit, it detects a busy channel and so must reschedule. Table 2.2 shows the updated event queue. Note the transmission attempt has been incremented from 0 to 1 to reflect this need to reschedule. Car 0 has chosen an offset time of 278.236 μs which is added to the current simulation time of 835.554 μs to generate the event time of 1,113.79 μs shown in Table 2.2. In this simulation scenario, the communication period is only 1 ms which means Car 0 has rescheduled the transmission outside the available period. As a result, Car 0 fails to transmit and its message is aborted.

The simulation time remains at 835.554 μs because Car 0 was unable to transmit during the communication period. Meanwhile, when Car 2 attempted its transmission at 799.280 μs, it detected a busy channel as well and so began the rescheduling procedure. Table 2.3 shows the results of the rescheduling. With a generated offset of 52.875 μs, Car 2 reattempts its transmission 888.429 μs into the simulation. This time is in the communication period and so is successful.

As presented in this section, a scheduled message can take three tracks. If the channel is sufficiently clear, the message will be successfully sent as scheduled like Car 1 in the example; however, if another transmission is occupying the channel or the previous transmission ends less than DIFS prior to the scheduled transmission, the message must be rescheduled. The rescheduled message may still be successful, as was the case for Car 2, or it may fall outside the communication period and be aborted such as Car 0. A fourth scenario may theoretically occur. When the communication protocol is implemented, hardware is needed to acquire access to the communication channel and to detect whether the channel is busy or idle. This hardware operates by shifting bits and hence requires a finite time before events can be detected. It is therefore possible that two vehicles will generate a sufficiently identical offset so as to both attempt transmitting without being aware of the other vehicle's transmission. As a result, the two transmission collide and neither is successful. Due to the high-speed processing available, it is reasonable to assume that channel acquisition occurs instantaneously, so for a packet collision to occur, the generated offset for each vehicle would need to be identical. Due to the extremely low probability of such an occurrence, packet collisions are not investigated in this work. Instead, it is assumed that all vehicles are instantaneously aware of a transmission.

Table 2.2: Vehicle 0 reschedules transmission out of bounds.

vehicle_number	event_type	transmission_attempt	event_time	initial_attempt
0	comm.	1	0.000748155	0.000748155
2	comm.	0	0.00079928	0.00079928

Table 2.3: Vehicle 1 successfully reschedules message.

vehicle_number	event_type	transmission_attempt	event_time	initial_attempt
2	comm.	1	0.00079928	0.00079928

2.4 Simulation Analysis

As a newly accepted amendment to the IEEE 802.11 family of standards, there is little established documentation on the performance of 802.11p available. The protocol set forth by 802.11p for the PHY and MAC layers is identical to 802.11a with the only modification being the available data rates [4,22]. This observation is advantageous because the prevalence of 802.11a has led to well documented models of the protocol's throughput - the amount of useable data sent through the channel in a given time window. By modifying the data rates, hence only changing the message time used in the communication model, a general representation of 802.11a can be developed. The throughput of the 802.11a model can then be determined and compared to the results of more detailed communication models.

Referring once again to Figure 2.3, which shows the message packet developed for 802.11p, the 16 μs to transmit the preamble, the 24-bit header, and the 192-bit data packet all remain the same for 802.11a. The only change is the time required to send the combined 216 bits of data. With a range in data rates from 6 Mbps to 54 Mbps, the total message time ranges from 52 μs to 20 μs, respectively. The throughput for each of these message times was calculated to place lower and upper bounds on the throughput for comparison to other models. For each message time, the number of vehicles was incremented and the throughput for each car count was averaged over 100 simulations to generate a meaningful statistical average.

To produce a plot of the lower bound throughput, the message time was set to 52 μs. The number of vehicles attempting to transmit during one communication period was incremented from 2 to 500. Once again, one hundred simulations were run at each vehicle count to generate a meaningful statistical average for the number of successful transmissions. Each transmission contains 192 bits of total data so the throughput, TP, can be determined

using:

$$TP = \frac{192 * AverageSuccessful}{T}, \tag{2.1}$$

where T is the communication period. The useful payload size is 96 bits per transmission because this is the amount of actual data supplied to the receiving vehicles. An identical process is used to generate an upper bound on the throughput once the message time is modified to 20 μs. Figure 2.4 summarizes the results of both simulations showing the upper and lower bounds for each communication period, T.

The results obtained from the simulation have the same functional form as those presented by Y. Xiao and J. Rosdahl for the throughput of 802.11a [19] which helps validate the simulation model. Figure 2.5 shows a reproduction of the upper throughput limits for 802.11a generated by Xiao and Rosdahl for three communication rates: 6 Mbps, 24 Mbps, and 54 Mbps. These curves have the general functional form:

$$y(x) = A(1 + Be^{Cx}), \tag{2.2}$$

where y represents the channel throughput in Mbps and x is the payload size in bytes.

Comparing Figure 2.4 to Figure 2.5 shows a variation in curve shape which is a function of differences in parameters A, B, and C of (2.2). There are several reasons for this difference. Most significantly, the throughput analysis by Xiao and Rosdahl did not incorporate a time out sequence. Rather than have a finite period in which a message had to be scheduled, their messages were rescheduled as many times as necessary until successful transmission was achieved. This allowed for a larger number of bytes to be delivered. Additionally, the payload was adjusted by increasing the data packet size while keeping a fixed number of transmission nodes. By contrast, the communication model maintained a fixed data size but increased the number of vehicles attempting to transmit. Adding more vehicles increased the total attempted payload size but also increased the message overhead because each vehicle also included an additional preamble and header in its transmission. These two factors culminated in a lower throughput for the communication model but this is not

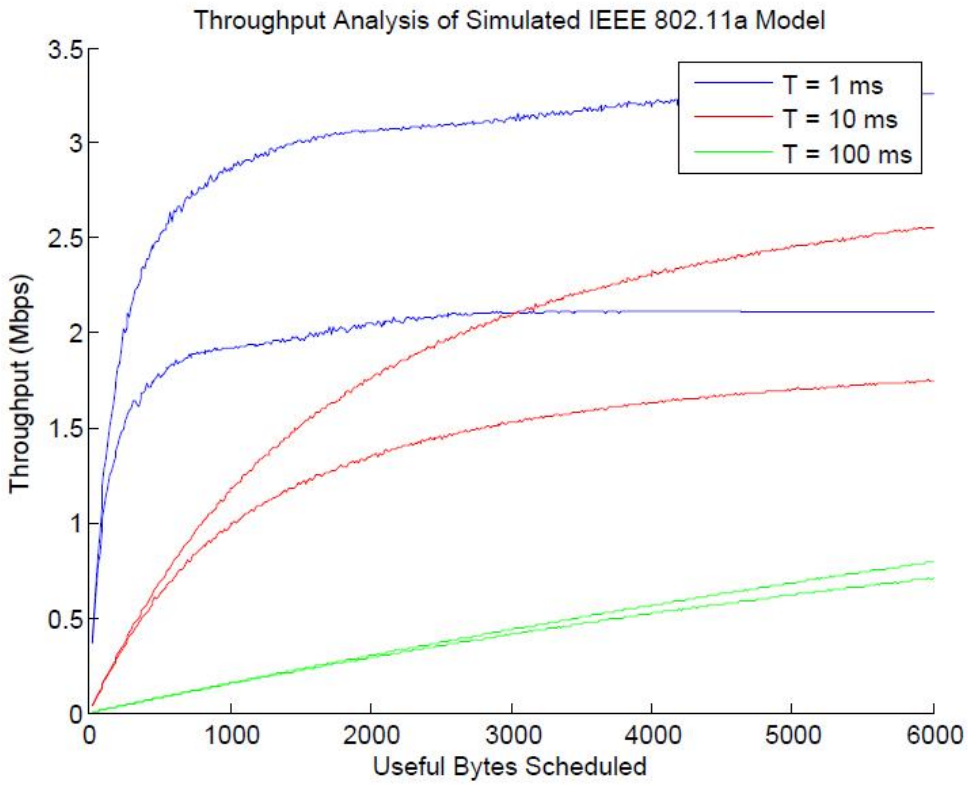


Fig. 2.4: Throughput results from communication model after modifying to resemble the IEEE 802.11a protocol.

unexpected. The same functional form remains encouraging.

The comparison of throughputs was done as validation for the communication model, and was successful, but the goal of the communication analysis remains to explore the interaction between platoon size and the communication protocol described by 802.11p. To this end, the effect of platoon size on transmission error rate was studied. This crucial piece begins characterizing how much information each controller will receive in a given time frame. As a first step, the model was readjusted to represent the 802.11p protocol by returning the message time to 88 μs. The number of aborted messages were recorded as platoon size was incremented from 2 to 50 vehicles. For this section, one thousand simulations were run for each car count to provide reasonable statistical analysis. Figure 2.6(a) summarizes the results of this analysis for three different communication periods. As expected, a larger communication period similarly allows for a larger platoon size. Looking toward the in-

tegration of communication with the control algorithms; however, a large communication period has downfalls as it corresponds to a larger delay between data updates. Nonetheless, a communication period of 10 ms comfortably allows for twenty-five vehicles while maintaining less than 10% error rate as shown in Figure 2.6(b) which zooms in on Figure 2.6(a). In this scenario, the error rate is purely a result of messages being rescheduled outside the initial communication period and does not reflect channel interference. The results are still encouraging, however, because they show from a scheduling perspective, a platoon of twenty-five vehicles can send kinematic information every 10 ms with 80% success rate.

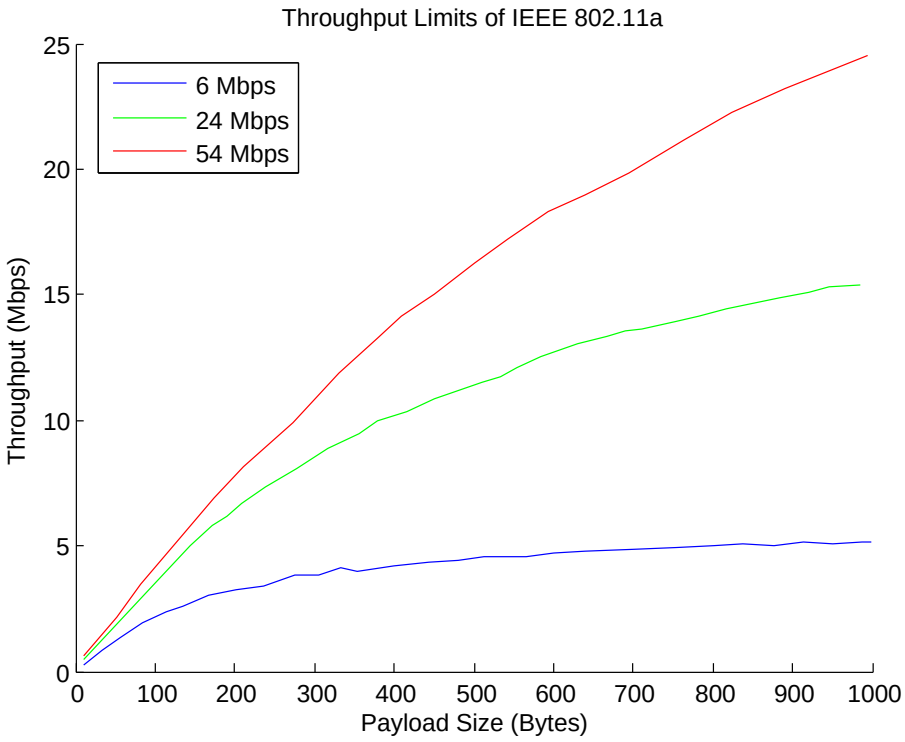
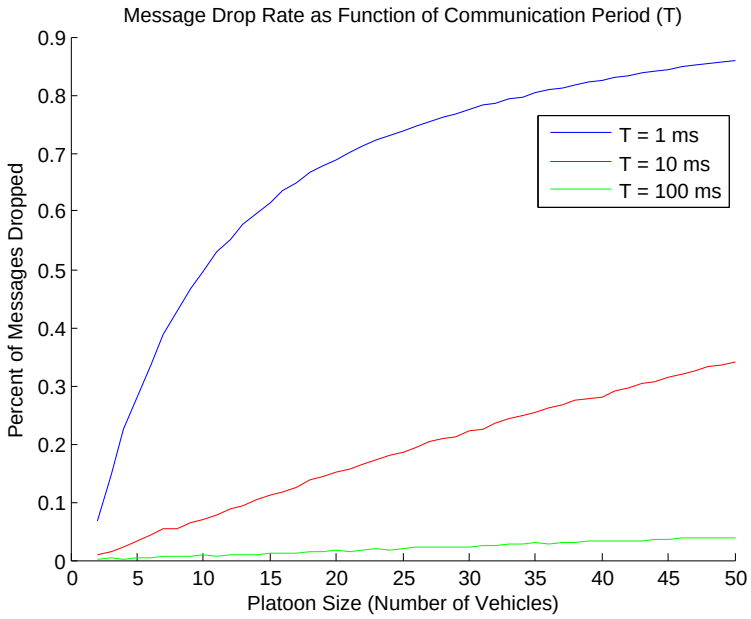
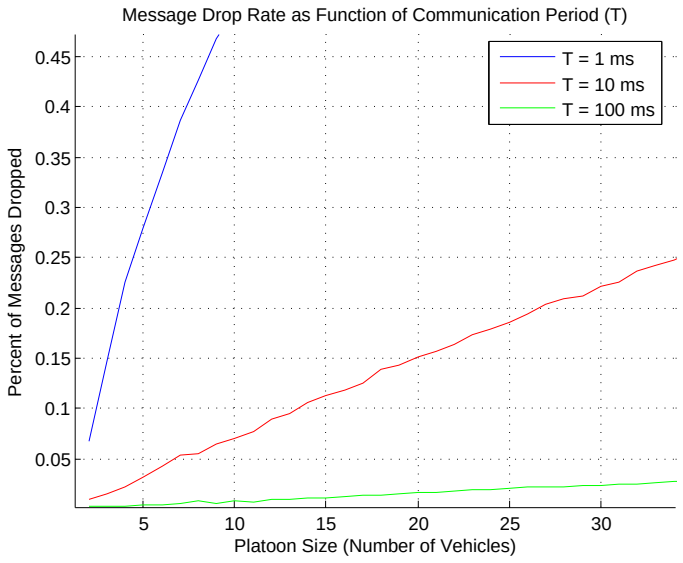


Fig. 2.5: Established upper throughput limits for 802.11a.



(a) Rate of unsuccessful transmissions as a function of platoon size for three different communication periods (T).



(b) Zooming in on the error rate associated with the 10 ms communication period.

Fig. 2.6: The transmission loss rate as a function of platoon size for three communication periods.

Chapter 3

Control Algorithm for Vehicle Platooning

To be fully implemented, vehicle platooning requires several levels of control. PATH broke the algorithms into two general categories: lateral control and longitudinal control. The lateral control is primarily responsible for modifying the steering wheel angle to maintain a vehicle's lane position. During the PATH field trials, the center of the lane was marked with small magnets. Fluxgate magnetometers on the vehicle then determined the vehicle's position within the lane. Deviations from center were measured and used in the lateral controller to determine a compensatory angle on the steering wheel. Using this process, vehicles within the eight car platoon had less than three inches of error [1]. The lateral controller can be expanded to address vehicles entering and exiting the platoon with the inclusion of additional sensors and, most likely, additional communication between vehicles. These additional features govern the evolution of a platoon over time but are not necessary for a static platoon, that is a platoon with a fixed number of members. This thesis presumes the platoon is nearly formed so no vehicles are merging. Vehicles are not necessarily efficiently or safely spaced, however. Under this assumption, the primary concern is achieving, then maintaining, a pre-determined ideal gap between vehicles. PATH termed this aspect of vehicle platooning "longitudinal control."

3.1 Ideal Vehicle Spacing

PATH and SARTRE used vehicle spacings of 6.5 m and 6 m during their respective demonstrations [1, 7]. The gap used by PATH was based on human reaction times although an autonomous vehicle will be more responsive. Therefore, from the perspective of an autonomous platoon, a 6.5 m spacing is more tolerant of variations in velocity and acceleration between vehicles but it also, potentially unnecessarily, decreases the improvement to road-

way capacity and reduces the aerodynamic benefits of platooning. Wind-tunnel tests at the University of Southern California revealed drag forces were cut in half when vehicles were spaced about half a car length apart compared to PATH's 6.5 m spacing. Analyses at University of California Riverside showed the reduced drag corresponded to 20-25% improvement in fuel economy [1]. NASCAR provides further evidence for the improved efficiency obtained from reducing vehicle spacing. A new technique, known as push drafting, emerged on the Talladega course in 2011 and led to improvements of nearly 15 mph. In push drafting, two cars link up bumper-to-bumper, providing twice the horsepower without significantly increasing the drag. Engineers from three different racing teams commented on the dramatic improvement provided by this tightly spaced platoon which essentially eliminates all drag on the following car [25]. Interestingly, one concern resulting from the vacuum is overheating of the second car's engine due to the lack of air flow. As a result, the second car is often slightly skewed from the first car. This provides some airflow to the engine at the cost of increased drag on the car.

Improvements in efficiency support decreasing vehicle spacing, but safety concerns also push for more tightly spaced platoons. The severity of a vehicle collision increases with the difference in velocities, the relative velocities, between the vehicles. In a platooning situation, the vehicles will be traveling at nearly identical velocities so the severity of a collision is relatively minimal unless an emergency braking scenario occurs. At this point, one vehicle brakes at maximum capacity, causing an abrupt decrease in velocity and maximizing the difference in velocities between itself and the proceeding vehicle. If enough space is between the braking vehicle and its follower, the second vehicle can initiate braking in an effort to match velocities. This behavior is exhibited by human drivers and leads to larger vehicle spacing. As mentioned, this spacing reduces efficiency and roadway capacity. On the other end of the spectrum in the emergency braking scenario, vehicles can be very closely spaced. When the front vehicle fully engages its brakes, the velocity decreases very little before the following vehicle collides. While this collision may occur at high speeds, the relative velocities are very minimal so the risk of severe damage to vehicles and passengers

is similarly minor.

By looking at incident data from California, a threshold difference in velocity of 3.3 m/s was observed wherein no data existed for lower velocities [26]. In other words, there was zero damage reported in collisions where vehicles were traveling within 3.3 m/s of each other. Building on this observation, a conservative difference of 2.5 m/s was used to determine acceptable vehicle spacing in the work by Jackson [5]. Note, federal regulations require vehicle bumpers to withstand collisions equivalent to a 5 mph (2.235 m/s) impact with an equivalently sized, parked vehicle [27]. The models developed revealed a low probability of damage in collisions with vehicle spacing on the order of 0.1 m when Monte Carlo simulations were run to introduce variation in braking ability, headway spacing, and initial velocities. For this particular work, more strict boundaries are placed on vehicles. It is assumed all vehicles have an identical maximum braking ability of 9.5 m/s^2 [5]. When the two vehicles collide, the difference in velocity upon collision can be determined using:

$$\Delta v = at = \sqrt{2(\Delta s)a}, \tag{3.1}$$

where "Δs" is the initial difference in position and "a" is the deceleration of the first car. Rearranging Eq. (3.1), the initial vehicle spacing can be found as:

$$\Delta s = (\Delta v)^2/2a. \tag{3.2}$$

Using Jackson's conservative estimate of 2.5 m/s and a maximum braking capacity of 9.5 m/s^2 leads to safe vehicle spacing of 0.57 m, significantly smaller than the 6 m spacing used in demonstrations. These results do not imply the 6 m spacing is unsafe because Eq. (3.1) and Eq. (3.2) assume the following vehicle does not attempt to brake. At 0.57 m, this is a reasonable assumption due to communication and actuation times in automatic systems. If the vehicles are traveling at 27 m/s (60 mph), they cross the gap of 0.57 m in 0.21 seconds. However, at 6 m the following vehicle should be aware the preceding vehicle is braking and so can begin reducing its velocity to mitigate collision damage.

For safe platooning that better improves efficiency and roadway capacity, this work will focus on implementing a control algorithm which is string stable and can maintain a vehicle spacing of 0.1 m. This vehicle spacing requirement is extremely rigid when compared to the calculated safe gap of 0.57 m, but the computer simulations used to evaluate the control algorithm are idealized. The exact braking capabilities of a vehicle are not accurately represented because they depend on the mechanical system of the car as well as roadway conditions. Actuation delay is also not incorporated into the model for similar reasons. To account for the influence of these factors, a stricter spacing requirement is used to provide some safety measure. It should be noted the results presented here and in proceeding chapters use an ideal gap of 0.1 m but, within the code itself, this vehicle spacing is an arbitrary constant and can easily be modified if data for different scenarios is desired.

3.2 Longitudinal Control

Deviations from the ideal gap are used to evaluate the stability of control algorithms. For the platoon itself to be stable, these deviations must decrease with each additional car. Early PATH work emphasized the need for inter-vehicle communication to guarantee string stability within the longitudinal control but did not stipulate what information vehicles needed to share. A large number of control algorithms were proposed in the late 1970s. Some algorithms attempted to mitigate vehicle nonlinearities [28, 29], while many others focused on maintaining consistent spacing [30–33]. Despite the theoretical benefits of each proposed controller, the ultimate goal of vehicle platooning is a physical implementation. Of the documented control algorithms, one has been field tested. The PATH eight-vehicle demonstration relied on a robust, sliding-mode control design developed by Rajesh Rajamani et al. [2].

$$\ddot{x}_i^* = (1 - C_1)\ddot{x}_{i-1} + C_1\ddot{x}_0 - (2\xi - C_1(\xi + \sqrt{\xi^2 - 1}))\omega_n\dot{\varepsilon}_i \tag{3.3}$$

$$-(\xi + \sqrt{\xi^2 - 1})\omega_n C_1(\dot{x}_i - \dot{x}_0) - \omega_n^2\varepsilon_i$$

As seen in Eq. (3.3), each vehicle i={1, 2, ..., N}, where N is the platoon size, uses

the velocity, $\dot{x}$, and acceleration, $\ddot{x}$, of the preceding, i-1, and lead, 0, vehicle to calculate a desired acceleration of $\ddot{x}^*$. The controller bandwidth is ω_n and was set as $1/(2\pi\mathrm{T})$ in this paper with T denoting the communication period although the controller was observed to be relatively robust to deviations in this value given precise kinematic data. The spacing error for the i^{th} vehicle is denoted by ε so that:

$$\varepsilon_i = x_i - x_{i-1} + L, \tag{3.4}$$

$$\dot{\varepsilon}_i = \dot{x}_i - \dot{x}_{i-1}, \tag{3.5}$$

where L is the ideal spacing and was set to 0.1 m for reasons presented earlier.

The remaining two variables in the control algorithm of Eq. (3.3) are C_1 and ξ. C_1 is a control gain in the interval [0, 1] that reflects the relative weighting of data from the lead car compared to the preceding car. For example, setting C_1 to 0 removes the lead vehicle's data from the calculation while a C_1 of 1 results in a control algorithm entirely dependent on the lead vehicle's data. The tuning of ξ is less prescriptive. As may be expected given the structure of Eq. (3.3), ξ can be viewed as a damping ratio and takes values in the interval [1, ∞) with 1 being critically damped.

3.3 Implementation of Longitudinal Control Algorithm

PATH ultimately demonstrated the longitudinal control's effectiveness at maintaining a vehicle spacing of 6 m in a controlled setting [1, 2]. The goal of the control simulation is to recreate the robust algorithm of Eq. (3.3) and explore the consequences of reducing the vehicle spacing to 0.1 m.

To simulate a platooning environment, the position, velocity, and acceleration of each vehicle at the current time are stored in an array. This array also includes the kinematic values which will be implemented at the beginning of the next period. The period in this simulation is directly connected to the communication period which was the focus of Chapter 2 because, ultimately, it is the time interval at which vehicles will receive the data

needed by the control algorithm. Recall, the lead vehicle is assumed to have information from the infrastructure regarding necessary velocity and acceleration values for the next period. All following vehicles rely on communicated velocity and acceleration values to determine a desired acceleration which will ensure the ideal gap is maintained between all vehicles in the platoon.

At the beginning of the simulation, the lead vehicle is defined to be at a position of 0. All vehicles are assumed to have negligible length. Although this assumption is unrealistic, the actual vehicle length will simply shift the spacing of following vehicles, complicating the presentation but not altering the results. The lead vehicle then receives velocity and acceleration values from the sub-routines "get_velocity" and "get_acceleration," respectively. The get_velocity subroutine is user-defined and intended to provide varying velocity data to the lead vehicle so the followers' responses can be observed. For this work, these subroutines each take the current time and return a velocity and acceleration, with units m/s and m/s^2, respectively. The get_velocity function is a simple sinusoid:

$$\dot{x}_0(t) = 20.0 + sin(t/\gamma), \tag{3.6}$$

where t is the current time and γ was initially equal to 5.0. The get_acceleration function uses Newton's difference quotient to approximate the time derivative of the velocity function from Eq. (3.6).

$$\ddot{x}_0(t) = \frac{\dot{x}_0(t+\delta) - \dot{x}_0(t-\delta)}{2\delta} \tag{3.7}$$

A time step of δ is used where δ was set to 1 μs. The lead vehicle's velocity and acceleration values at the start of the simulation are also stored as the next velocity and next acceleration to complete the lead vehicle's initialization.

Having a user-defined velocity function for the lead vehicle is similar to the current highway structure wherein speed limits are posted. The get_acceleration function follows directly from the get_velocity function through numeric derivation. Although acceleration plays more of a background role for human drivers, it is necessary to directly incorporate

the acceleration in this simulation for two reasons. Two vehicles can have identical velocity at one point in time, with ideal spacing between them, but if the acceleration of the rear car exceeds that of the lead car, a collision will occur. Therefore, it is important to ensure vehicles within the platoon are matching accelerations as well as velocities. This necessity is reflected in the controller's need for accelerations as seen in Eq. (3.3). Acceleration is also important in this simulation because it is ultimately the control signal for the longitudinal controller. The amount the gas pedal is compressed or released, and the duration of that pedal position, changes a vehicle's acceleration. Human drivers currently adjust the acceleration to achieve a desired velocity. The longitudinal controller replaces the driver and so must make its own adjustments of the acceleration. Furthermore, physical limits on acceleration must be incorporated into the simulation to assess the feasibility of control algorithms. Large accelerations are not physically possible nor, from a physiological perspective, desirable. For this work, it is assumed passengers and vehicles can comfortably obtain a maximum acceleration of 3 m/s^2. Similarly, a comfortable deceleration is set to 3 m/s^2. Larger decelerations can be used in an emergency braking scenario but such situations are not explored in this work. If the lead vehicle's resulting acceleration exceeds 3 m/s^2, the acceleration is instead set to the bound.

Once the lead car has received velocity and acceleration information according to Eqs. (3.6) and (3.7), each following vehicle is initialized. The following vehicles begin with the same velocity and same acceleration as the lead vehicle but the initial position is a function of each vehicle's relative position within the platoon. For the preliminary investigation, it was assumed vehicles were ideally spaced. Under this assumption, the position of each vehicle is determined in a straightforward manner using:

$$x_i(0) = x_0(0) - (i * L). \tag{3.8}$$

Note, Eq. (3.8) assumes negligible vehicle length so ideal_gap is 0.1 m for this work. If a non-zero vehicle length were used, that length could simply be added to the ideal_gap term though this change would merely result in a shift of the following vehicles' positions but

would not alter the analysis or results which focus on the gap spacings.

With all the vehicles' kinematic data initialized, the simulation begins. The lead vehicle determines its next velocity value using its current velocity and acceleration according to:

$$\dot{x}_i(T_r) = \dot{x}_i(t) + \ddot{x}_i(t) * T_r, \tag{3.9}$$

where i=0 for the lead vehicle and T_r is a user-defined variable to account for actuation and processing delays. T_r can be considered a vehicle's reaction time and allows for more realistic simulations. For all simulations in this work, T_r was set equal to 1 ms or 1/10th of the period. The lead vehicle predicts its position at the end of the period based on its current position, velocity, and acceleration in a similar manner according to the basic motion law described by:

$$x_i(T_r) = x_i(t) + \dot{x}_i(t) * T_r + \frac{1}{2}\ddot{x}_i(t) * T_r^2, \tag{3.10}$$

with i=0 once again.

All following vehicles track the lead vehicle's motion but also continue moving based on the old parameters for T_r. After T_r, each following vehicle will have a "next_velocity" and "next_position" defined by Eq. (3.9) and Eq. (3.10), respectively. Based on their position and velocity after T_r, the following vehicles then calculate the acceleration needed for the remaining time period, R, to achieve the ideal_gap per Eq. (3.3). Initially, the communication is assumed to have zero dropped or distorted packets so the controller can access the preceding and lead vehicles' velocity and acceleration data. The important consequences of dropped or distorted packets are discussed in Chapter 4.

Each following vehicle calculates its next acceleration directly according to Eq. (3.3). The velocity and acceleration data used in the formula are the values stored as "next_velocity" and "next_acceleration" by the respective vehicle. The control algorithm in no way restricts the desired acceleration so to ensure realistic behavior, the acceleration and deceleration are bounded.

At this point, the current simulation time is still zero (t=0). Every vehicle remains at its initial position with the velocity and acceleration parameters assigned during initialization. Based on these preliminary values, each vehicle has now calculated its next_position, next_velocity, and next_acceleration. These "next" parameters are what each vehicle will have at a time of T_r and govern the behavior of that vehicle for the remaining duration of the period, R. In other words,

$$R = T - T_r; \tag{3.11}$$

so at the end of the period, each following vehicle will have position, velocity, and acceleration given by:

$$x_i(T) = x_i(T_r) + \dot{x}_i(T_r) * R + \frac{1}{2}\ddot{x}_i(T_r) * R^2, \tag{3.12}$$

$$\dot{x}_i(T) = \dot{x}_i(T_r) + \ddot{x}_i(T_r) * R, \tag{3.13}$$

$$\ddot{x}_i(T) = \ddot{x}_i(T_r). \tag{3.14}$$

Meanwhile, the lead car also moves forward, updating its kinematic data. Its position at the end of the next period is a function of its current position as well as its initial velocity and acceleration. Specifically:

$$x_0(T) = x_0(t) + \dot{x}_0(t) * T + \frac{1}{2}\ddot{x}_0(t) * T^2, \tag{3.15}$$

where t=0 for the first period. Note, this is a different update than for the following vehicles because it uses the initial kinematic values rather than the predicted values. Unlike the following vehicles, the lead vehicle's reaction time, T_r, is not incorporated until the next time iteration. The velocity and acceleration are calculated from Eq. (3.6) and Eq. (3.7), respectively, with the next acceleration also being set to the value returned from Eq. (3.7).

After proceeding through the updates described from Eqs. (3.9)-(3.15), every vehicle has moved through the first period and the current time is equal to T. Vehicle motions for the remaining simulation time follow the same updating scheme every period. A total of 5000 periods comprised each simulation. The current position, velocity, and acceleration

for every vehicle is stored for analysis while a visual output is displayed in the computer console. The visual display simply uses ASCII pound symbols to represent following vehicles while an asterisk denotes the lead vehicle, allowing for a quick visual of how vehicles are spaced and whether oscillations are occurring in the vehicles' positions.

3.4 Validation of Initial Control Algorithm Simulation

The primary validation of the initial control algorithm is it achieves ideal vehicle spacing. Using an update period (T) of 10 ms and a ten-vehicle platoon, all simulations were ran over 5000 periods. When the following vehicles were initially spaced at ideal_gap behind their predecessor and the lead vehicle followed the varying velocity trajectory described in Eq. (3.6) with γ=5.0, the maximum spacing error is less than 0.001 m while differences in the vehicle velocities are on the order of 10^{-4} m/s. Increasing the initial vehicle spacing to 0.2 m, or 2*ideal_gap, results in the relative vehicle spacing shown in Figure 3.1. Vehicles converge to the ideal_gap in under 5 s and maintain this spacing very consistently through the remainder of the simulation with a few deviations less than 0.001 m in amplitude.

Using γ equal to 5.0 in Eq. (3.6) produced a qualitatively gentle velocity trajectory. It served as a base evaluation of the control algorithm in Eq. (3.3) and shows the anticipated behavior of a ten-vehicle platoon. Modifying the value of γ within the get_velocity subroutine explores the effect of more frequent velocity changes on platoon stability. Changing γ from 5.0 to 1.0 increases the frequency of speed variation. When ideally spaced upon initialization, following vehicles are still able to successfully match the lead vehicles velocity with differences on the order of 10^{-4} m/s. The spacing error also remains bounded by 0.001 m but the error never fully settles.

Increasing the initial vehicle spacing to 0.2 m reveals the control algorithm is still able to quickly drive the vehicle spacing toward the desired ideal_gap, see Figure 3.2. The following vehicles reach a spacing of 0.1 m in under 5 s, refer to Figure 3.2(a) which is nearly indiscernible from Figure 3.1, but oscillations of magnitude 0.001 m remain apparent for the remainder of the simulation time and are more frequent than when γ=5.0 as shown in Figure 3.2(b). The analysis presented by Rajamani et al. regarding the controller of

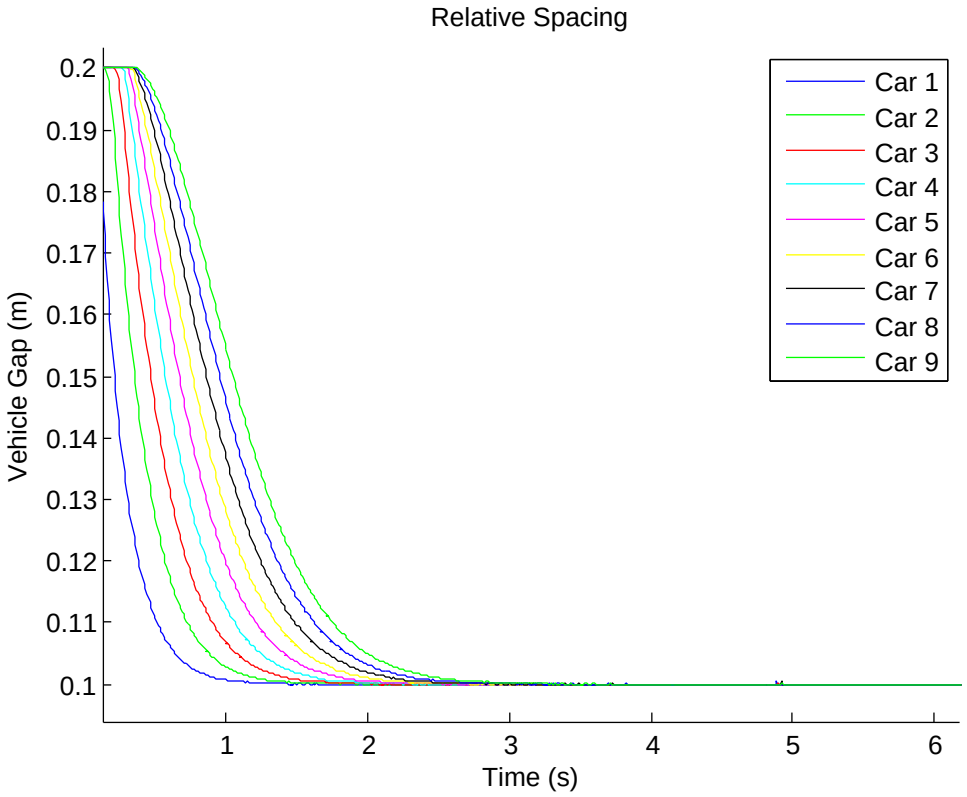
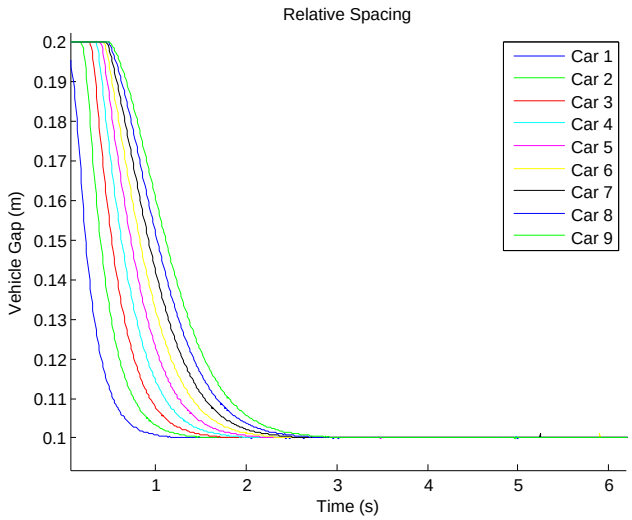


Fig. 3.1: Relative spacing between each following vehicle and its predecessor when initially spaced at 0.2 m.

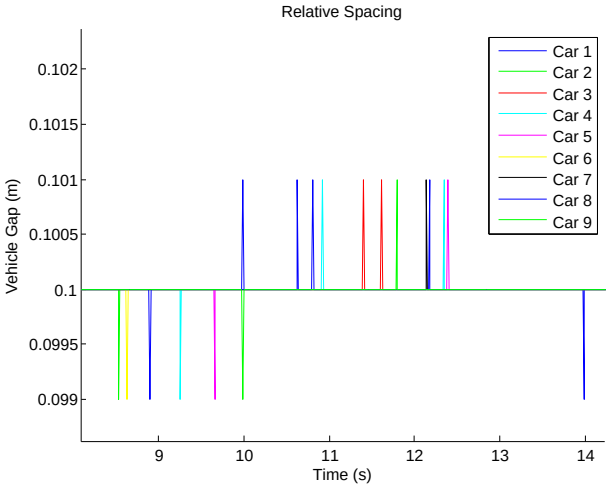
Eq. (3.3) also noted the occurrence of oscillations in vehicle positioning. Over the 7.6 mile course traveled for the PATH demonstration, vehicles remained within 0.2 m of the desired 6.5 m vehicle spacing [2].

Not surprisingly, the simulations also reveal oscillatory behavior in the vehicle velocities. The relative velocities between vehicles contained periodic oscillations on the order of 10^{-4} m/s following an initial convergence approximately 5 s into the simulation as shown in Figure 3.3.

Modifying γ in Eq. (3.6) so the sinusoidal trajectory has a period of 7.5 s then 5 s, with γ=0.75 and γ=0.5, respectively, further increases the frequency of the lead vehicles sinusoidal trajectory. For each new value of γ, the analysis focuses on the steady-state oscillations and so it is assumed all vehicles are initially ideally spaced. As expected, the



(a) Convergence of vehicle spacing to an ideal gap of 0.1 m.



(b) After settling, spacing errors remain bounded by 0.001 m.

Fig. 3.2: Relative spacing when vehicles are initially spaced at 0.2 m and follow a velocity trajectory of Eq. (3.6) with γ=1.0.

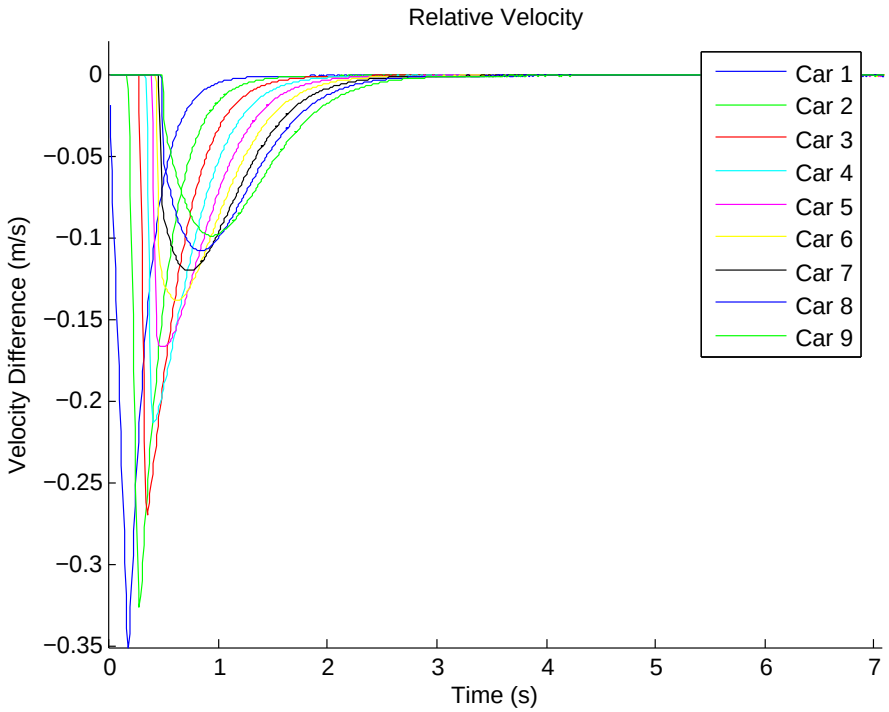


Fig. 3.3: Relative velocity between vehicles when initially spaced at 0.2 m and following a lead vehicle with velocity trajectory described by Eq. (3.6) with γ=1.0.

rate of oscillations in both the relative vehicle spacing and relative velocity increased with increasing frequency in the stipulated velocity trajectory. The magnitude of the oscillations, however, does not increase. The spacing oscillations remain bounded by 0.001 m when vehicles were initially spaced at the ideal_gap and the lead vehicle was tracking a sinusoidal velocity with a 5 s period. Meanwhile, the velocity oscillations do not exceed a magnitude of 10^{-4} m/s for the same scenario.

The magnitude of the oscillations in the relative vehicle spacing is sufficiently small compared to the ideal_gap so no collisions directly result from this behavior. These oscillations do reduce the platoon's efficiency, however. The scale of oscillations, both for positions and velocities, within the simulation are small but their presence is reflected in the physical PATH demonstration.

Intuitively, repeated braking and accelerating reduces vehicle fuel efficiency. The exact impact of this behavior is a function of the vehicle's mass, the amount of energy converted

to forward motion, the magnitude of speed variation, etc. As a minimal estimate, it can be assumed all energy from fuel is converted directly to increasing the vehicle's kinetic energy while decreases in velocity are achieved through braking or frictional loss. Therefore, in the computer simulation, a measure of the minimal energy expenditure for each vehicle can be calculated with Algorithm (3.1). Each vehicle will have a term E denoting the relative energy, with units J/kg, expended through the simulation.

Algorithm 3.1 Energy calculation algorithm

```
for (car=0; car<platoon_size; car++)
{
  if (v_new(car)> v_old(car))
  {
    E_car=E_car+(v_new^2 (car) -v_old^2 (car))
  }
}
```

Algorithm (3.1) provides a measure of the oscillations within the simulation, and hence a comparative measure of oscillations in a physical platoon. Thus far, four velocity trajectories have been explored by varying the value of γ in Eq. (3.6) systematically from 5.0 to 1.0, 0.75, then 0.5. The relative energy for each vehicle within the platoon was investigated for each trajectory when vehicles are initially ideally spaced. Ideal spacing was chosen because it is desirable to reduce the steady-state oscillations. Naturally, placing vehicles at twice the ideal_gap results in the last vehicle having the largest energy expenditure because that vehicle must overcome the largest distance for all vehicles to reach ideal spacing.

The oscillations cause each vehicle to expend more energy than its predecessor. Figure 3.4 shows the relative energy expenditure when the leader follows each of the trajectories explored. The relative energy is obtained by subtracting the lead vehicle's generalized energy, which is assumed to be the amount required to travel the given trajectory, from each following vehicle's relative energy expenditure. It is worth emphasizing the relative energy being shown in Figure 3.4 and referenced in this text is a generalized energy and has units J/kg. It merely provides a measure of the oscillations for the purpose of comparison.

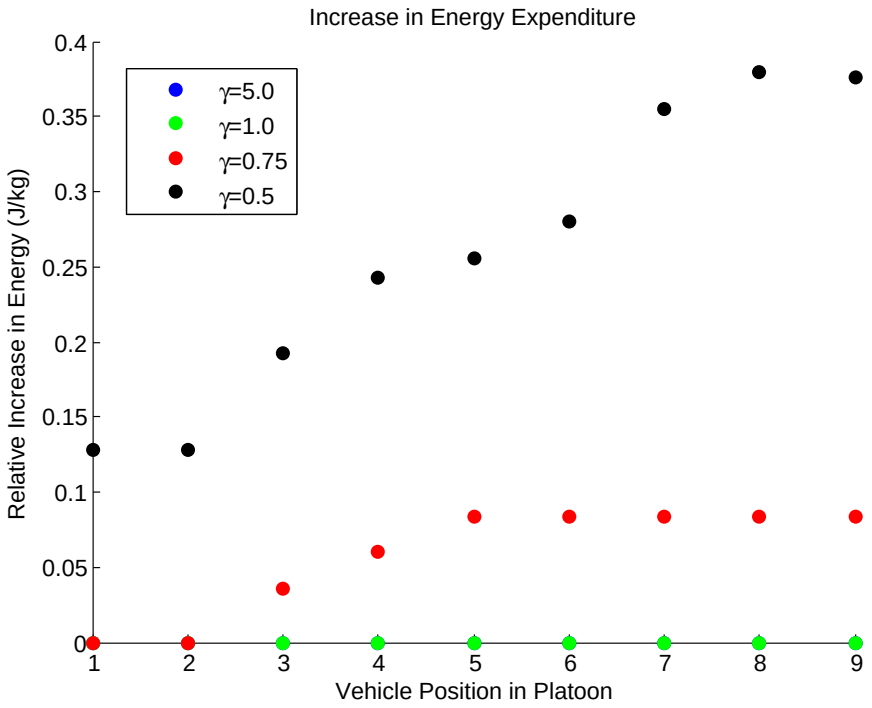


Fig. 3.4: Ideal relative energy expenditure for platoon with each following vehicle spending nearly the same energy and less than the lead vehicle.

When following the baseline trajectory of Eq. (3.6), no oscillations were observed in the vehicles' relative positions or velocities. As a result, all vehicles use nearly identical energy resulting in the horizontal line denoted by γ=5.0 in Figure 3.4. By contrast, minor oscillations were observed when γ was modified to 0.75 and more frequent oscillations occurred at γ=0.5. In both these scenarios, each vehicle's relative energy expenditure exceed its predecessor's. The amount of increase appears linear for the first five platoon vehicles but then tapers off, though an increase in energy is still occurring.

One of the motivations for implementing vehicle platoons is to increase fuel efficiency but as Figure 3.4 reveals, oscillations in the following vehicles can reduce these benefits. Oscillations also pose a possible safety threat. Although the spacing errors within the simulation were an order of magnitude smaller than the ideal_gap and did not directly result in a collision, the model did not account for all vehicle behavior. Inertia and momentum within the physical system could potentially amplify oscillations in vehicle positions to the

point of a collision. In the PATH demonstration, the spacing error was 0.2 m which was acceptable when desired vehicle spacing was 6.5 m but will not be acceptable when the ideal_gap is reduced to 0.1 m [1].

When the desired acceleration of platoon vehicles is calculated using the control algorithm of Eq. (3.3), vehicles initially spaced at 0.2 m are able to converge to an ideal_gap of 0.1 m within 5 s. Vehicles maintain this spacing within ± 0.001 m while following a variety of sinusoidal velocity trajectories. Variations in velocity are on the order of 10^{-4} m/s. When the lead vehicle follows the trajectory described in Eq. (3.6), following vehicle positions and velocities settle to the desired values. As the frequency of the trajectory increases, oscillations within the following vehicles' positions and velocities appear and similarly increase. Within the simplified simulation, it is difficult to fully quantify the effect of these oscillations on the platoon; however, Figure 3.4 shows a generalized energy to quantify the influence of oscillations. The appearance of larger oscillations in the physical implementation show these simulated oscillations are a real concern. Modifying the control algorithm to reduce these oscillations will improve the efficiency benefits of platooning and reduce the possibility of a collision within the physical system. Chapter 4 will introduce an additional term to the control algorithm and show the accompanying reduction in oscillations.

Chapter 4

Combining Communication and Control Models for Vehicle Platooning

When transmission times were adjusted to match those of IEEE 802.11a, the communication model developed and presented in Chapter 2 produced throughput rates with the same functional form as the established throughput limits presented by Xiao and Rosdahl [19]. Returning the transmission times to model IEEE 802.11p and running simulations to increase the number of platoon vehicles scheduling transmissions at different communication periods revealed a 10 ms communication period provided enough time for 25 vehicles to transmit with less than 20% loss rate. Meanwhile, Chapter 3 investigated the performance of the control algorithm:

$$\ddot{x}_i^* = (1 - C_1)\ddot{x}_{i-1} + C_1\ddot{x}_0 - (2\xi - C_1(\xi + \sqrt{\xi^2 - 1}))\omega_n\dot{\varepsilon}_i \tag{4.1}$$

$$-(\xi + \sqrt{\xi^2 - 1})\omega_n C_1(\dot{x}_i - \dot{x}_0) - \omega_n^2\varepsilon_i,$$

which was implemented in the 1997 eight-vehicle PATH demonstration [2]. Simulations of these algorithm conducted by PATH prior to the demonstration had the algorithm successfully take a platoon from rest to a constant speed and then back to rest. Providing the lead vehicle with four sinusoidal velocity trajectories, a more severe test than undertaken by PATH, and assuming all following vehicles received accurate information from the lead vehicle and preceding vehicle within a 10 ms period, the spacing error and relative velocities were recorded. When the lead vehicle followed the qualitatively gentle trajectory:

$$\dot{x}_0(t) = 20 + sin(t/\gamma), \tag{4.2}$$

with γ=5.0, following vehicles initially spaced 0.2 m behind their predecessor converged to

an ideal_gap of 0.1 m within 5 s and maintained this spacing with minimal deviation in both position and velocity. Modifying the velocity trajectory by setting γ to 1.0 introduced oscillations in the relative spacing and velocity of the following vehicles. Decreasing γ to 0.75, and then to 0.5, increased the frequency of these observed oscillations. Introducing a relative energy term to quantify the effect of the oscillations revealed each vehicle generally required more energy than its predecessor. The difference was most pronounced with γ=0.5.

4.1 Analysis of Control Algorithm Robustness in Presence of Dropped Packets

The analysis conducted in Chapter 2 did not include vehicle control, while Chapter 3 assumed ideal communication. Individually, the models developed in these two preceding chapters provide useful platforms to investigate vehicle platooning, and provide results similar to the above summarizations, but greater insight comes from a more realistic model which combines communication and control. To develop this hybrid model, an array "success" was introduced into the C++ control code to document which vehicles successfully transmitted during each period. The array is initially a 1xN, all-zero array where N is the platoon size. Each column corresponds to the vehicle position, i.e. column two records the transmission success for vehicle two in the platoon.

Prior to moving at the beginning of each 10 ms period, the simulation jumps to the "packet_success" sub-routine. Within this sub-routine, each vehicle in the platoon randomly generates a number called "test." If test is greater than a user-defined "error_rate," the vehicle was able to successfully transmit a data packet during the period. This successful transmission is represented by placing a "1" into the success array for the vehicle. Recall the simulation assumes the lead vehicle receives information directly from the infrastructure so it experiences no communication loss; however, the lead vehicle does not necessarily successfully transmit information to the following vehicles as it adheres to the WAVE protocol which uses a contention-based channel acquisition scheme.

After the success array has been formed, following vehicles begin moving through the period. Each following vehicle calculates its "next_velocity" and "next_position" as described in the previous chapter but the vehicle's "next_acceleration" depends on the success

array. For a following vehicle to determine a desired acceleration for the next period per Eq. (4.1), both the lead vehicle and preceding vehicle must successfully transmit their velocity and acceleration. Hence, the success array must contain a "1" in the lead vehicle array-element as well as in the preceding vehicle array-element. If the sum of these two columns within success is greater than one, both the leader and predecessor successfully transmitted so the vehicle can calculate a desired acceleration using Eq. (4.1). By contrast, if the sum of the two success columns is less than or equal to one it indicates at least one of the necessary vehicles failed to transmit. In this scenario, the vehicle maintains its current acceleration.

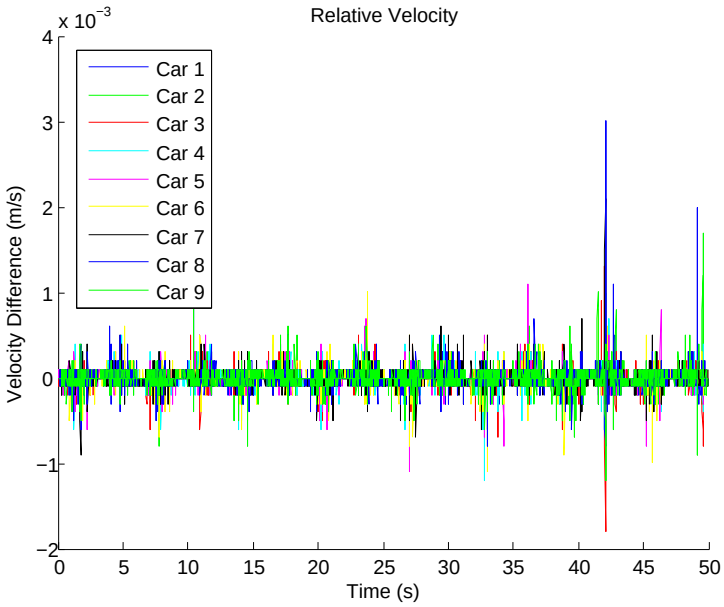
The user-defined error_rate was increased from 0% to 30% at 10% intervals. The same analysis from Chapter 3 was repeated at each error_rate. When the error_rate is 0%, every vehicle is able to successfully transmit and the results were identical to those obtained in Chapter 3, as expected, and serve as a baseline. Increasing the error_rate to 10% did not significantly alter the general platoon performance when the lead vehicle followed the initial velocity trajectory of Eq. (4.2) with γ=5.0. Although one in ten scheduled messages was unsuccessful, vehicles still converged to the ideal_gap in under 5 s after an initial spacing of twice the ideal_gap. No oscillations in the relative velocity were observed so no difference in relative energy occurred when compared to the corresponding trajectory in the base scenario.

Modifying the lead vehicle's velocity trajectory by settting γ equal to 1.0 reveals some consequences of dropped messages. The relative velocity increased by an order of magnitude as shown in Figure 4.1(a). The largest difference in velocity occurred near the end of the simulation, refer to Figure 4.1(b), and reached a magnitude of 0.003 m/s. This additional variation in velocity corresponds to an increase in the energy expenditure for each vehicle. Under the ideal communication assumption, every vehicle required the same amount of energy so the relative energy remained at 0 J/kg when the lead vehicle's velocity varied according to Eq. (4.2) with γ equal to one. By contrast, introducing a 10% loss rate resulted in Vehicle 8 requiring over 0.083 J/kg for the same trajectory over 50 seconds.

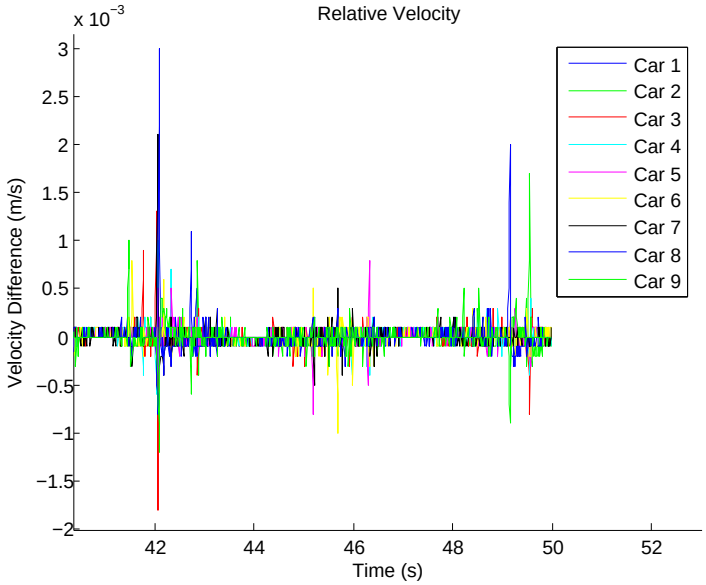
Increasing the frequency of the lead vehicle's sinusoidal velocity trajectory by setting γ equal to 0.75, then 0.5, further emphasized the increase in energy required by vehicles as a result of the 10% communication error_rate.

Figure 4.2(a) re-summarizes the relative energy expenditure for each following vehicle under ideal communication while Figure 4.2(b) presents the relative energy used when 10% of transmissions were unsuccessful. With ideal communication, each following vehicle expended more energy than its predecessor in a very linear fashion, but Figure 4.2(b) shows a more chaotic relationship between a vehicle's position within the platoon and the relative energy required. The scattered nature of the γ=0.5 trajectory in Figure 4.2(b) illustrates the susceptibility of platoon stability to random transmission losses. The internal modes of oscillation in the platoon are being excited by the random change in accelerations.

The relative energies shown in Figure 4.2(b) are more scattered than the ideal communication scenario as a result of accumulated behavior when a vehicle consistently fails to transmit. A failed transmission by one vehicle has a ripple effect on all following vehicles. By way of example, consider three consecutive periods. Assume the lead vehicle began decelerating in the first period and continues that acceleration through the next two periods. Vehicle 3 was unable to communicate during the first period. As a result, Vehicle 4 does not change its acceleration for the second period. Moving forward to the second period, Vehicle 4 now experiences increasing velocities while preceding vehicles are slowing down. The variations in velocity were small enough not to have a large impact on the vehicle spacing error but did contribute to a higher relative energy. The higher velocity enacted by Vehicle 4 during the second period is shared with Vehicle 5. Combining Vehicle 4's less than ideal kinematic data with true kinematic information from the lead vehicle, Vehicle 5's control algorithm produced desired accelerations to offset the errors of Vehicle 4. Vehicle 5's adjustments will be made in the third period but are already shared with Vehicle 6 which is able to more closely match the efficient trajectory of the lead vehicle and achieve a lower relative energy expenditure. Due to transmission losses from Vehicle 6, however, the corrected trajectory was not shared with Vehicle 7. As a result, Vehicle 7 also maintained

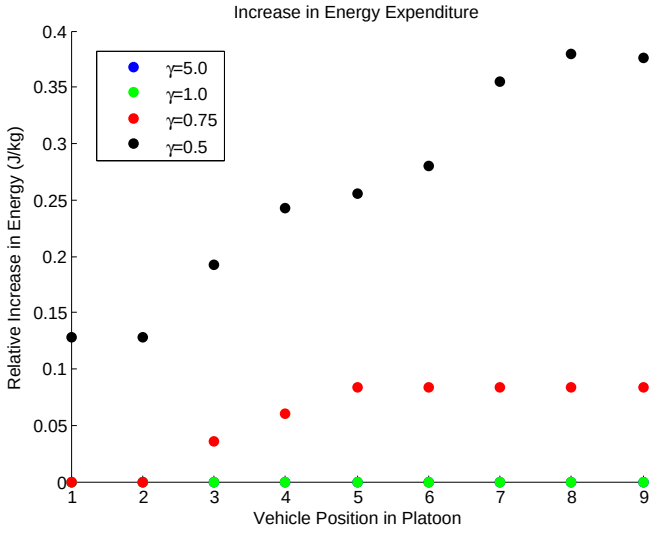


(a) The relative velocity increases when a 10% drop rate is modeled.

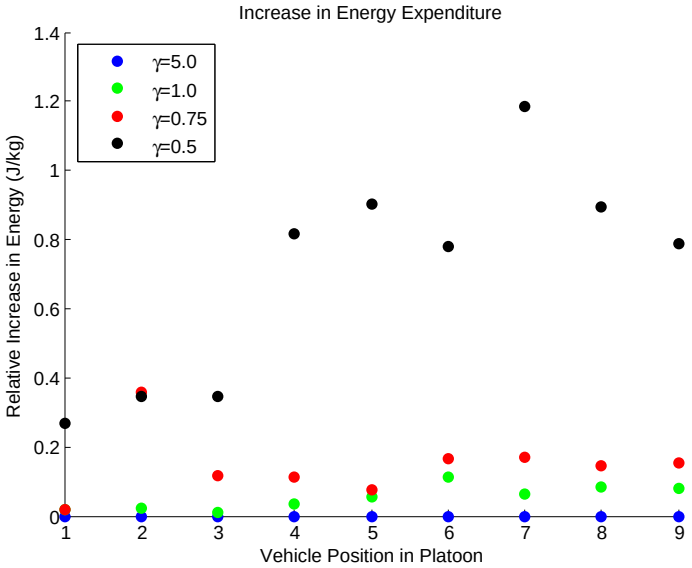


(b) Zooming in on the relative velocity shows a maximum difference of 0.003 m/s.

Fig. 4.1: Relative velocity when 10% of transmissions are unsuccessful and the lead vehicles velocity trajectory described by Eq. (4.2) with γ=1.0.



(a) Ideal communication.



(b) Transmission error rate of 10%.

Fig. 4.2: Relative energy (J/kg) expenditure for following vehicles when lead vehicle follows velocity trajectories described by Eq. (4.2) with γ=5.0, 1.0, 0.75, then 0.5.

a higher velocity than necessary. Vehicles 7 and 8 both transmitted successfully so each subsequent vehicle was able to approach the lead vehicle's trajectory and hence reduce the relative energy consumption.

The scenario described above is a sample behavior for three consecutive periods. Each simulation is comprised of 5000 periods, a time of 50 s, where communication success is a random probability. There were some periods wherein all vehicles successfully transmitted or when other vehicles failed to transmit. The sample period described is the dominant behavior however and results in the energy expenditures pictured for γ=0.5 in Figure 4.2(b). The velocity variations which produced the relative energy expenditures are shown in Figure 4.3. Note, the variation in speeds during each period.

With 10% of attempted transmissions unsuccessful, the ten-vehicle platoon experienced more oscillations and expended more energy than the ideal communication scenario. This

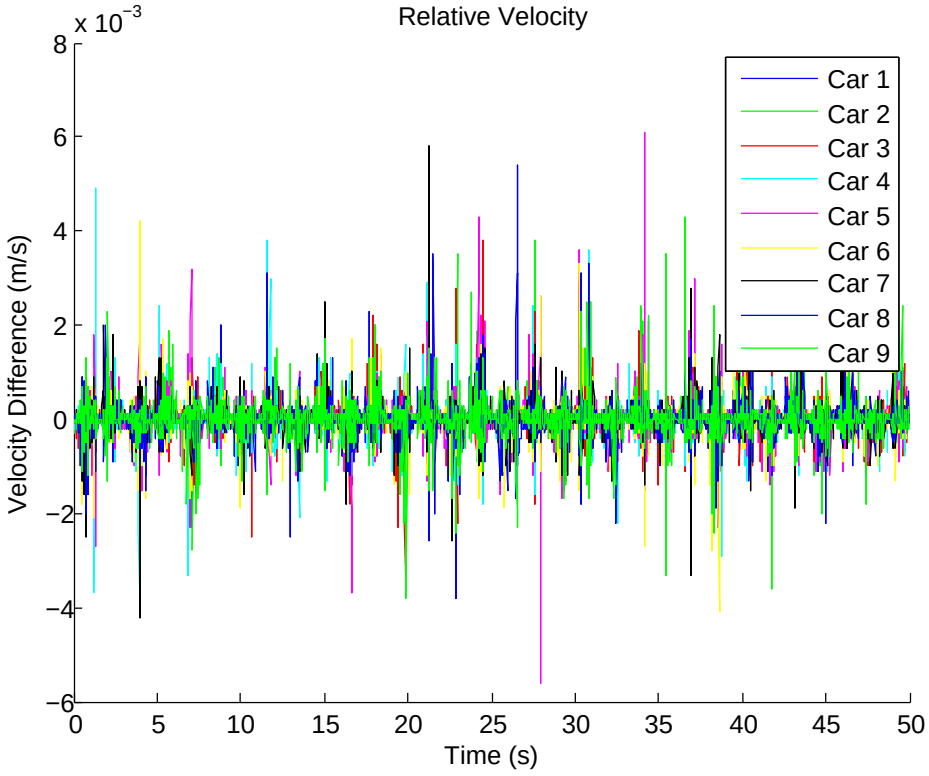


Fig. 4.3: Failed transmissions cause vehicles to have noticeably sharper changes in velocity.

deterioration in efficiency becomes more pronounced as fewer transmissions are successful. Adjusting the simulation to model a 20% transmission error rate increased from 0 J/kg with the ideal communication model while following a lead vehicle velocity trajectory of Eq. (4.2) with a γ of 5.0 to over 0.042 J/kg during the 50 second simulation. When γ was at 0.5, the maximum energy expenditure was over 4.8 J/kg at the 20% transmission loss rate, as shown in Figure 4.4, compared to under 0.38 J/kg for the ideal case. The distribution of relative energy expenditure is larger so partially obscures the energy deviations occurring from transmission losses which distorted Figure 4.2(b) but the loss of up-to-date kinematic data is still quite apparent, particularly for the γ=1.0 scenario.

Previous scenarios contained oscillatory behavior, but the vehicle spacing consistently remained within 0.001 m of the ideal_gap during all four trajectories. Losing 20% of the transmissions caused more variable spacing error. When the lead vehicle followed the gentle

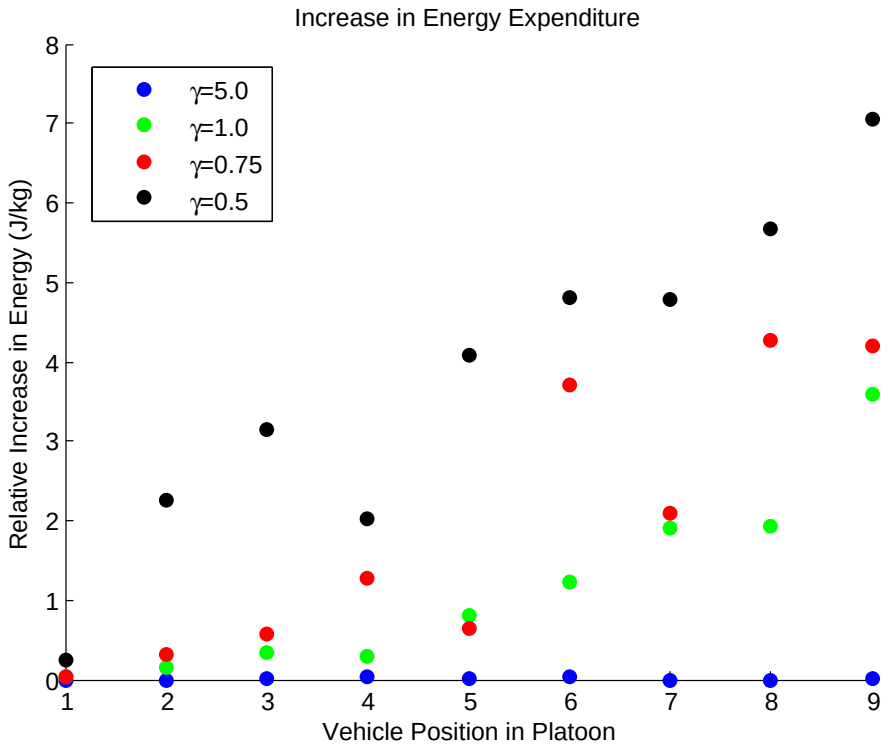
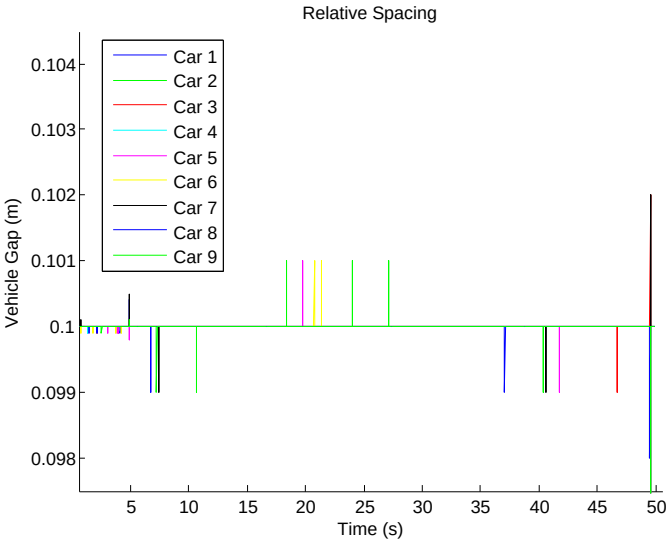


Fig. 4.4: Losing 20% of the transmissions increased the energy expenditure for every trajectory with each vehicle using more energy than its predecessor.

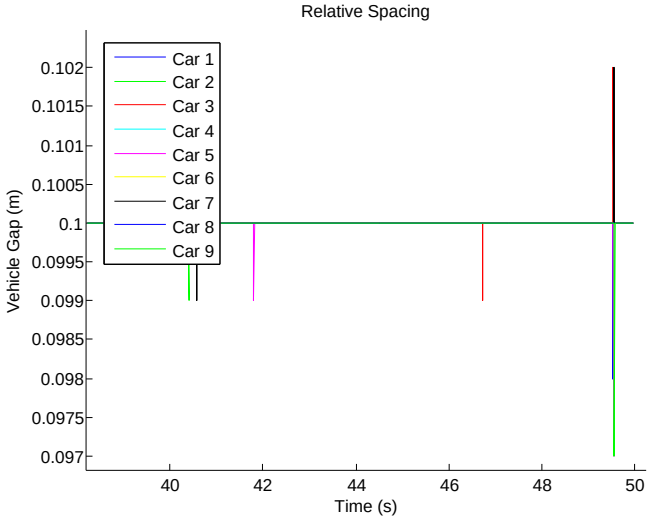
trajectory of Eq. (4.2) with γ=5.0, primary steady-state spacing errors generally remained bounded by 0.001 m but a maximum spacing error of 0.003 m was experienced by Vehicle 9 while Vehicle 3 had an error of 0.002 m during one period. The relative spacing for the entire simulation is shown in Figure 4.5(a). A closer look at the maximum spacing error is shown in Figure 4.5(b). Surprisingly, nearly all vehicle spacing errors remained bounded by 0.001 m when the lead vehicle tracked the sinusoidal velocity described by Eq. (4.2) with γ at the minimum value of 0.5. As shown in Figure 4.6, vehicles experienced frequent spacing errors but only Vehicle 2 experienced a momentary error greater than 0.001 m. It is worth noting, when vehicles were initially spaced at twice the ideal_gap, all ten vehicles still reached general convergence though, with γ equal to 0.5, approximately 7 s were required for the platoon to settle.

Corresponding to the increased spacing error and larger relative energy expenditures, vehicles experienced larger velocity differences in all four trajectories. Under ideal communication, the relative velocity remained on the order of 10^{-4} m/s with the trajectory only affecting the frequency of oscillation. Losing one in five transmissions resulted in relative velocities orders of magnitude larger. Following the slow-varying trajectory given by Eq. (4.2) with γ=5.0 still resulted in occasional velocity differences five times larger than previous scenarios. In the most extreme instance, a γ value of 0.5 caused several vehicles to experience relative velocities exceeding 0.03 m/s as presented in Figure 4.7.

The overall platoon performance deteriorated noticeably when the transmission error rate was increased to 30%. When following vehicles were initially spaced 0.2 m behind their predecessors, transmission losses caused the spacing error to reach as much as 0.02 m while tracing Eq. (4.2) with γ=5.0. Unlike previous simulations, the high communication loss led the control algorithm to consistently generate desired accelerations which exceeded the ± 3 m/s^2 acceleration bounds placed in the simulation as shown in Figure 4.8. Changing the lead vehicle's velocity trajectory by decreasing values of γ further amplified errors within the platoon. When γ was set to 0.5 and vehicles were initially spaced 0.2 m behind their predecessor, the control algorithm successfully drew vehicles to within 0.004 m of the



(a) Relative spacing over entire simulation.



(b) Maximum spacing error.

Fig. 4.5: Following the velocity trajectory of Eq. (4.2) with γ=0.5 and a 20% transmission loss rate resulted in larger spacing errors than previous scenarios wherein all oscillations remained within $\pm$0.001 m of the ideal_gap which is 0.1 m.

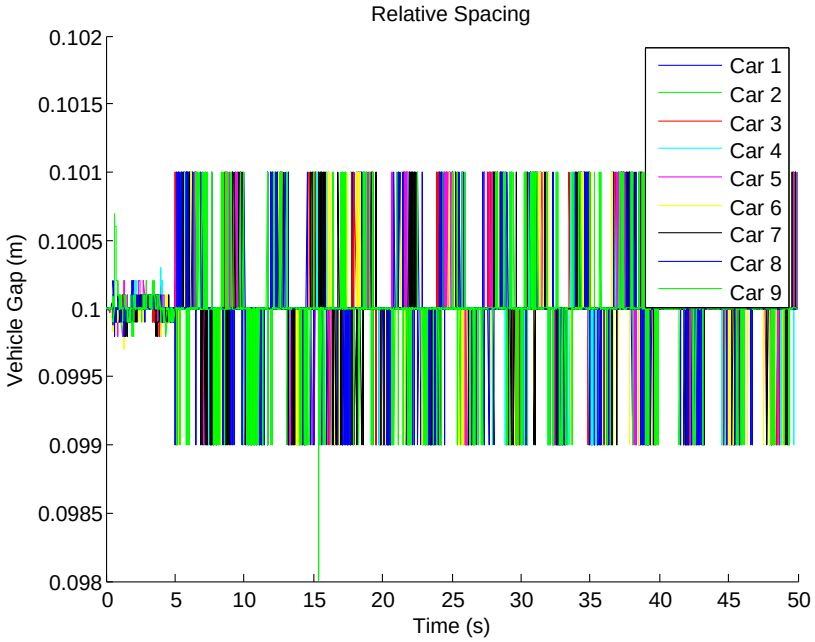


Fig. 4.6: Spacing errors remained bounded by 0.001 m when 20% of transmissions are lost and the lead vehicle follows Eq. (4.2) with γ=0.5.

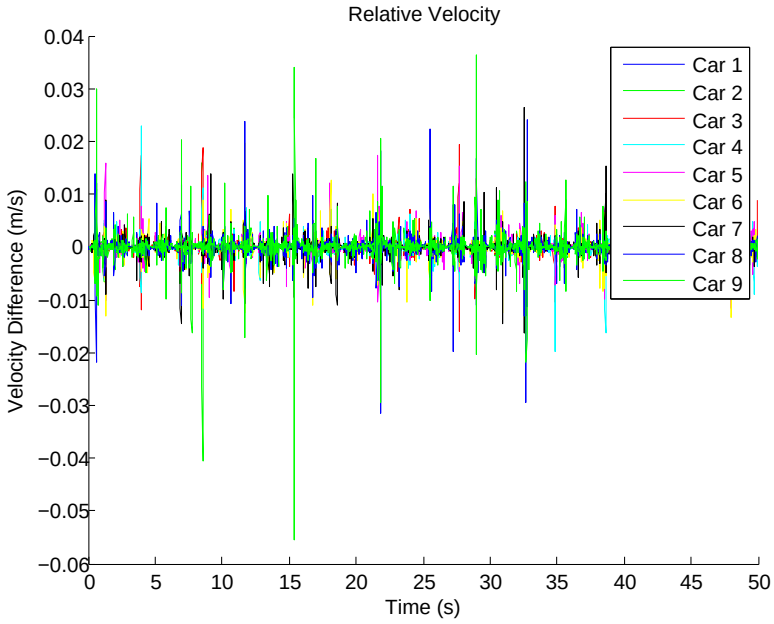


Fig. 4.7: Changing the trajectory to Eq. (4.2) with γ=0.5 increased the occurrence of velocity differences greater than 0.03 m/s.

ideal_gap in 5 s but spacing errors began growing throughout the rest of the simulation. This increasing error is visible in Figure 4.9. The poor platoon performance is further reflected in the large relative energy used by each vehicle as summarized in Figure 4.10. As with the majority of previous communication error rates, the rearmost vehicle requires the most energy requiring over 240 J/kg more than the lead vehicle during the 50 s simulation. Relative energy expenditures did not exceed 0.4 J/kg when all transmissions were successful.

Unlike previous error rates, a 30% transmission loss produced increasing spacing errors, as noted in Figure 4.9. In a physical implementation, this could result in vehicle collisions and hence is unacceptable so a lower communication error rate must be achieved. Further, the energy expenditures required to compensate for a 30% transmission loss rate are over 30 times larger than those at 20% loss rate.

Chapter 2's analysis of the WAVE protocol found a communication period of 10 ms accommodated a platoon of 25 vehicles with less than 20% error rate. Control algorithm

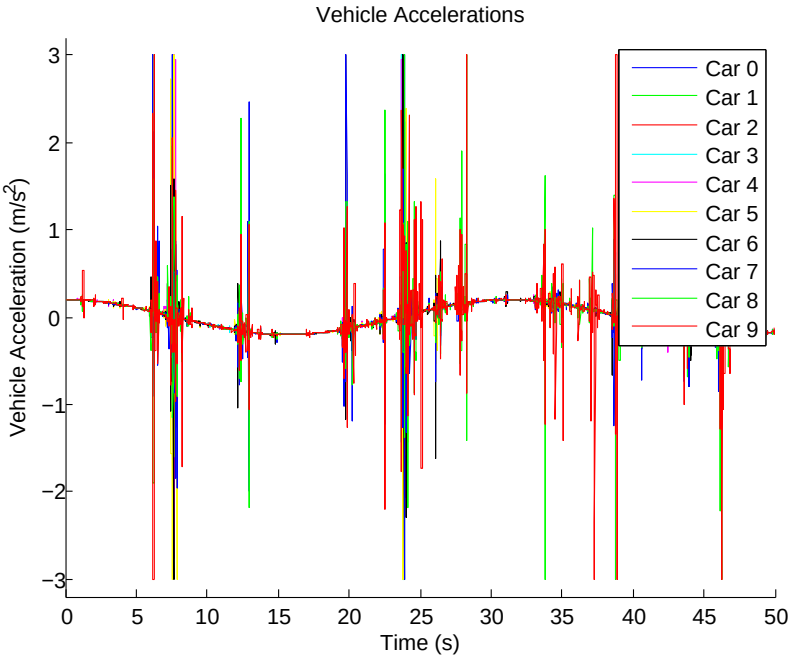


Fig. 4.8: High transmission loss caused the controller to generate desired accelerations exceeding the physical bounds of the simulation, forcing railing.

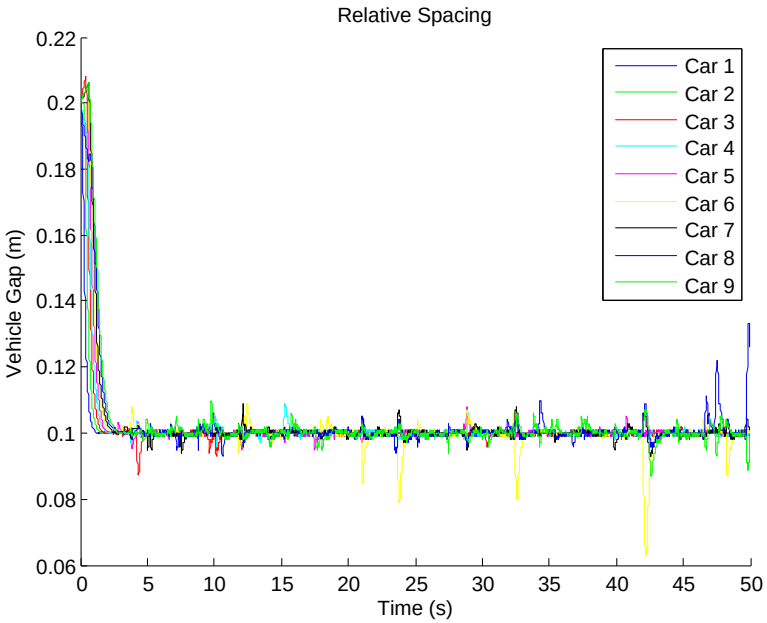


Fig. 4.9: When γ=0.5 in Eq. (4.2), vehicles initially spaced at 0.2 m approached the ideal_gap but experienced increasing spacing error.

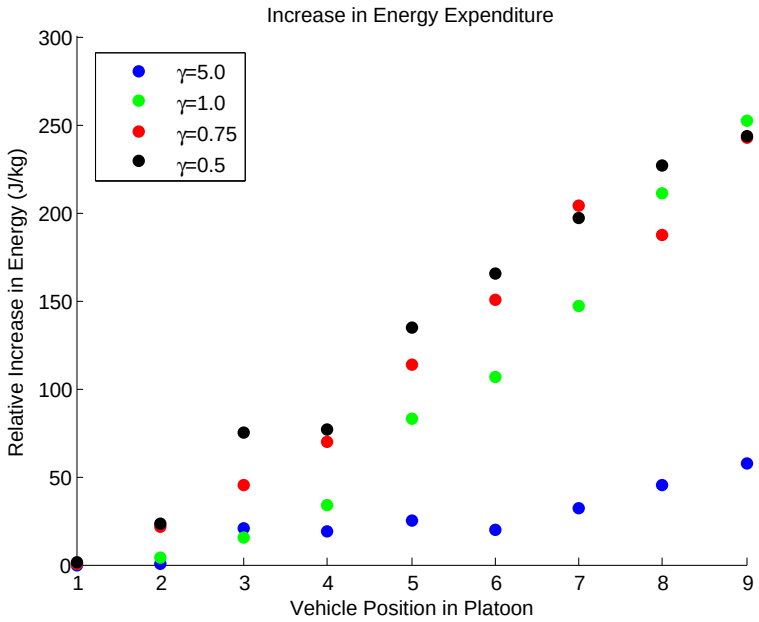


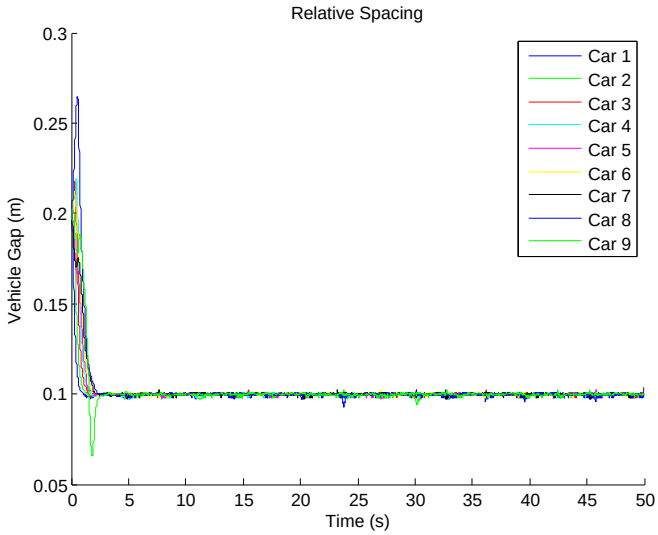
Fig. 4.10: Large energy expenditures were required for all four trajectories when a 30% transmission loss rate is implemented.

simulations with a 20% transmission error rate resulted in spacing errors on the order of 0.002 m which provides a comfortable margin of safety when the ideal_gap is 0.1 m. In this scenario, variations in vehicle velocity caused the relative energy expenditure to be larger than in the ideal communication scenario, however. When the lead vehicle followed Eq. (4.2) with γ equal to 0.5, the rearmost vehicle expended just over 7.06 J/kg in 50 seconds at the 20% error rate compared to under 0.04 J/kg when all transmissions were successful. Unsuccessful transmissions lead to higher energy requirements because vehicles over-compensate. A similar vehicle response is observed when vehicles receive imprecise kinematic data from other platoon members.

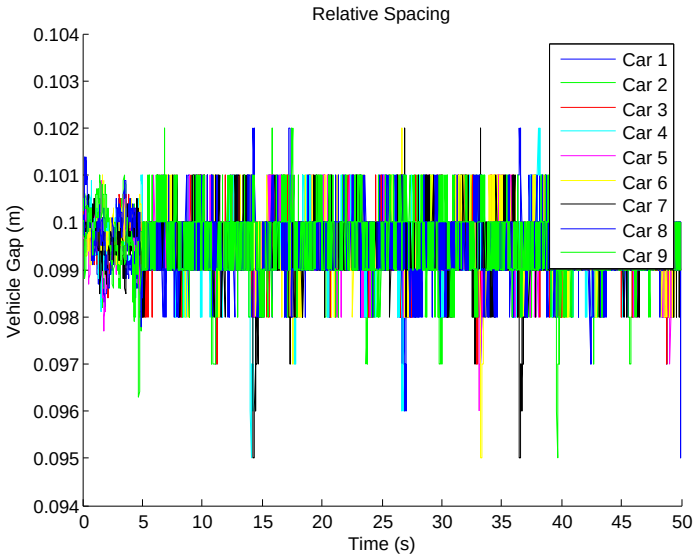
4.2 Introducing Kinematic Variation

Thus far, it has been assumed all transmissions contain exact information, within double precision, so only the transmission error rate effected efficiency and platoon stability. In reality, the kinematic data reported by each vehicle will be measured by sensors which have limited accuracy and are susceptible to variation. Measured data will therefore have some deviation from the true value which vehicles will need to tolerate. To model this realistic attribute, a normally distributed random value, herein referred to as variation, is added to the true kinematic value. The variation is characterized by a standard deviation, σ, which slightly alters the velocity and acceleration values shared with other platoon vehicles. For simulations in this section, ideal communication was assumed to isolate the influence of variation on platoon efficiency. All figures in the remainder of this chapter are not statistical but represent individual simulations unless otherwise noted.

Using a small σ of 0.01 did not significantly alter the previously observed bounds on vehicle spacing but did greatly impact both the relative velocities and hence generalized energy expenditure of each vehicle. As shown in Figure 4.11, vehicles were still able to converge to the ideal_gap when initially spaced at 0.2 m and remain within 0.005 m of ideal_gap while following the lead vehicle's trajectory of Eq. (4.2) with γ=0.5, the strictest trajectory explored.



(a) Vehicles converged to the ideal_gap within 5 s when initially spaced 0.2 m behind their predecessor.



(b) Steady-state spacing error.

Fig. 4.11: Relative vehicle spacing while tracking the velocity trajectory of Eq. (4.2) with γ=0.5, ideal communication, and variation with magnitude 0.01 standard deviation.

There was minimal oscillation in the relative velocity when vehicles followed a trajectory described by Eq. (4.2), γ=0.5, with ideal communication and no variation, see Figure 4.12(a). The oscillations which did occur remained on the order of 10^{-4} m/s, even when transmission losses were incorporated. Introducing kinematic variation with standard deviation of magnitude 0.01 resulted in relative velocities exceeding 0.04 m for all vehicles as shown in Figure 4.12(b).

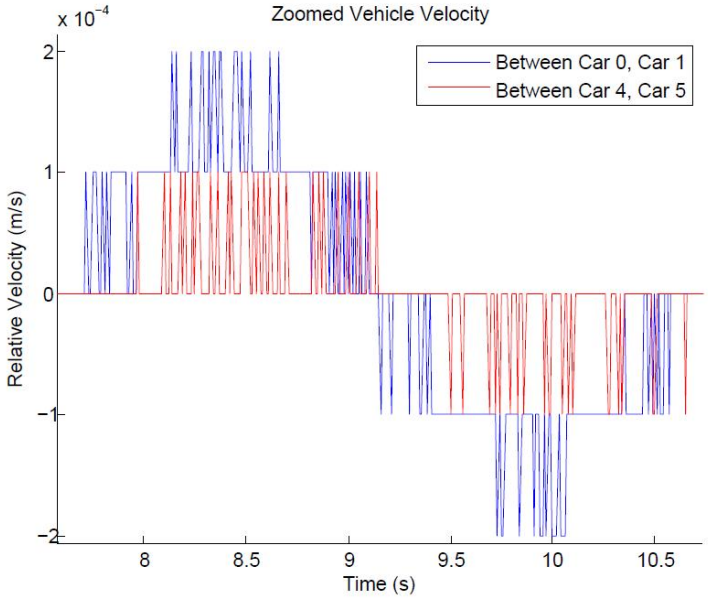
Increasing the standard deviation of variation to a magnitude of 0.02 caused frequent oscillations in the spacing error. The steady state spacing error was generally bounded by 0.005 m though two vehicles experienced 0.01 m spacing errors near the end of the simulation as shown in Figure 4.13. As anticipated, the magnitude of velocity also increased with frequent oscillations occurring with magnitude greater than 0.1 m. The relative velocity is shown in Figure 4.14.

Large energy expenditures accompanied the increase in relative velocity when variation was incorporated into the model. Under ideal conditions with the largest value of γ, each vehicle's energy expenditure for the 50 s simulation was on the order of 100 J/kg. By comparison, incorporating a variation with σ=0.01 increased this energy expenditure by a factor of ten so each vehicle required over 1000 J/kg for the same trajectory. As a measure of this large energy increase resulting from kinematic variation, the generalized platoon energy was calculated for each value of σ tested, specifically σ=0.0, 0.01, and 0.02, by summing each member's generalized energy. Table 4.1 summarizes the results of this summation.

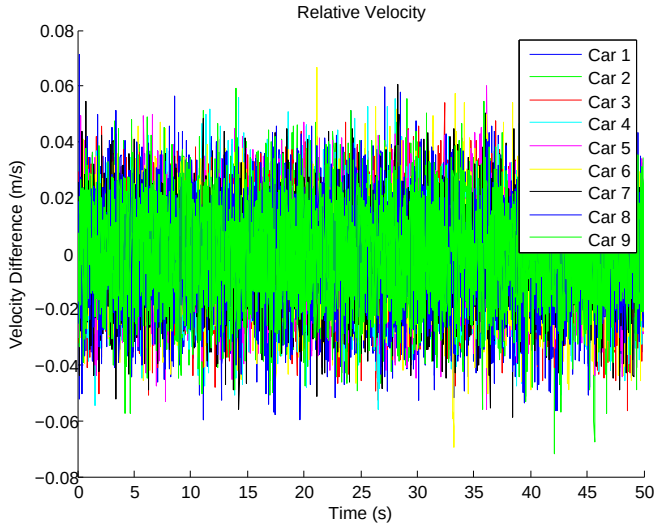
While the individual spacing errors still seem small, within 10% of the ideal spacing and comparable to the dropped packet scenarios, the collective behavior in the platoon was interesting. The platoon length decreases as the standard deviation of the transmitted

Table 4.1: Relative energy expenditure (J/kg) as a function of kinematic variation.

	σ=0.0	σ=0.01	σ=0.02
γ=5.0	1209.2	11984.34	21526.37
γ=1.0	6288.422	13087.19	21798.87
γ=0.75	8405.993	13710.04	22476.78
γ=0.5	12591.67	16405.10	23728.19

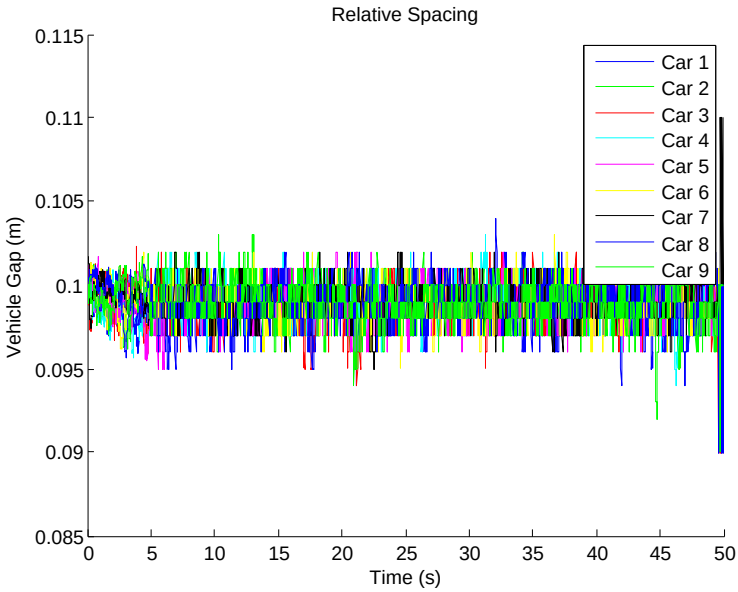


(a) Relative velocity without kinematic variation.

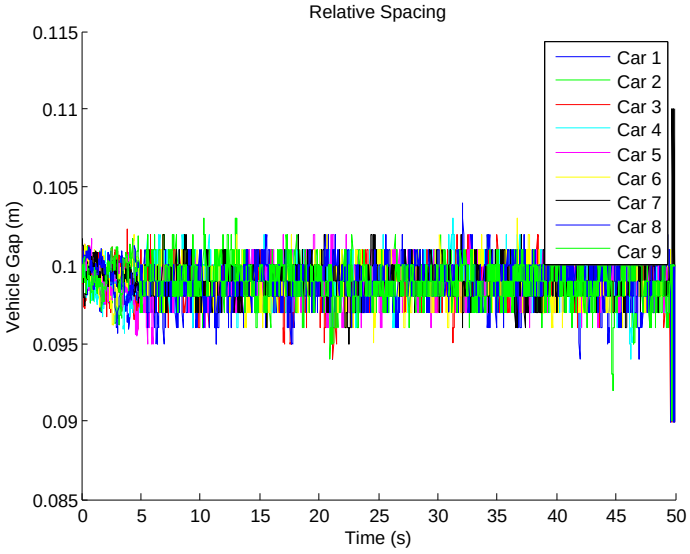


(b) Relative velocity when variation with standard deviation of magnitude 0.01 was introduced.

Fig. 4.12: Relative velocities required to follow the velocity trajectory of Eq. (4.2) with γ=0.5 when all communicated information was successfully transmitted.



(a) Relative spacing primarily within 0.005 m of ideal_gap.



(b) Vehicles did experience a maximum spacing error of 0.01 m near the end of the simulation.

Fig. 4.13: Relative vehicle spacing, assuming ideal communication, while tracking Eq. (4.2) with γ=5.0.

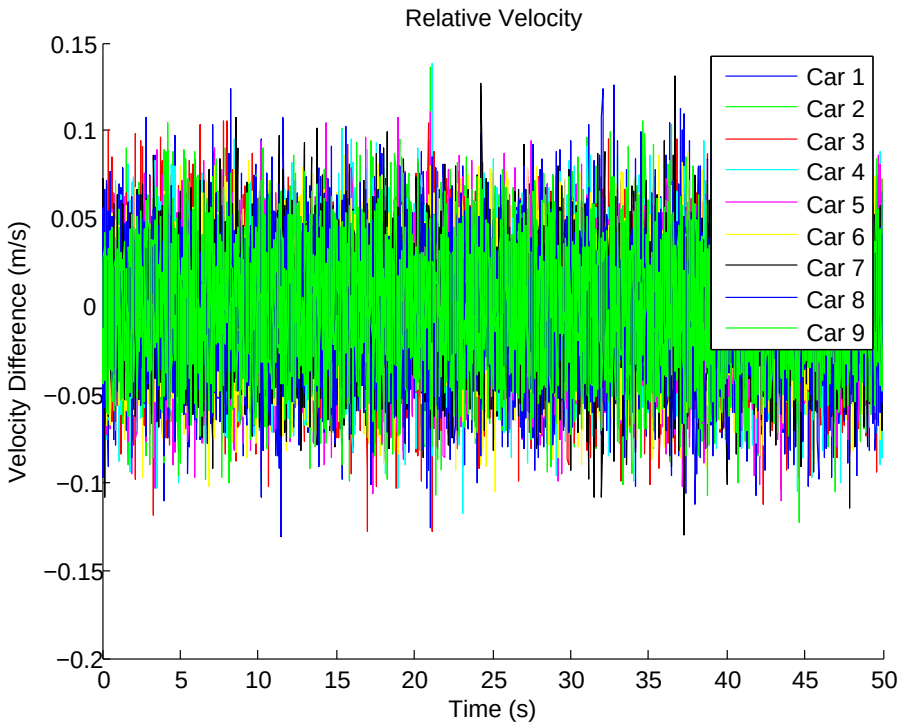


Fig. 4.14: A variation with σ=0.02 produced large velocity oscillations when vehicles tracked Eq. (4.2) with γ=5.0.

kinematics is increased. To study this undesirable platoon behavior, the simulation was ran using a ten vehicle platoon with vehicle spacing at ideal_gap. The platoon follows a constant velocity trajectory of 20 m/s. Under these conditions, the impact of the standard deviation in the transmitted data on the overall average platoon length is apparent as shown in Figure 4.15. The maximum standard deviation of 0.04 m/s for a vehicle moving at 20 m/s translates into the vehicle being between a speed of 19.96 m/s and 20.04 m/s 67% of the time. Yet, this error, on the order of 8 cm/s for each vehicle, and a corresponding error range in the transmitted accelerations cause the platoon length to shrink to 0.7835 m when the platoon length should be 0.9 m.

Further, as shown in Figure 4.16, the platoon exhibits low frequency oscillations after reducing its length. Intuitively, the observed oscillations are related to the amount of rapid change in acceleration. A Fast Fourier Transform of the data confirms the frequencies in

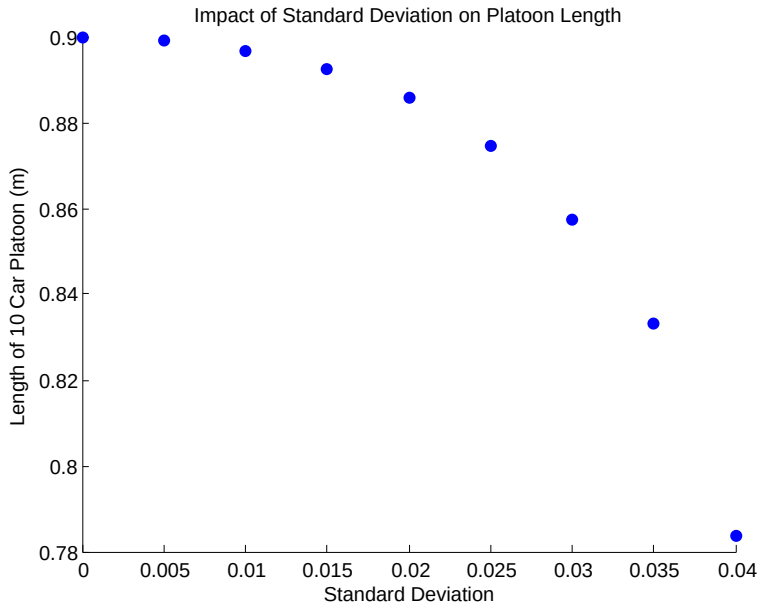


Fig. 4.15: When vehicles travel at a constant 20 m/s and are ideally spaced, the introduction of a standard deviation to transmitted data causes the overall platoon length to shrink.

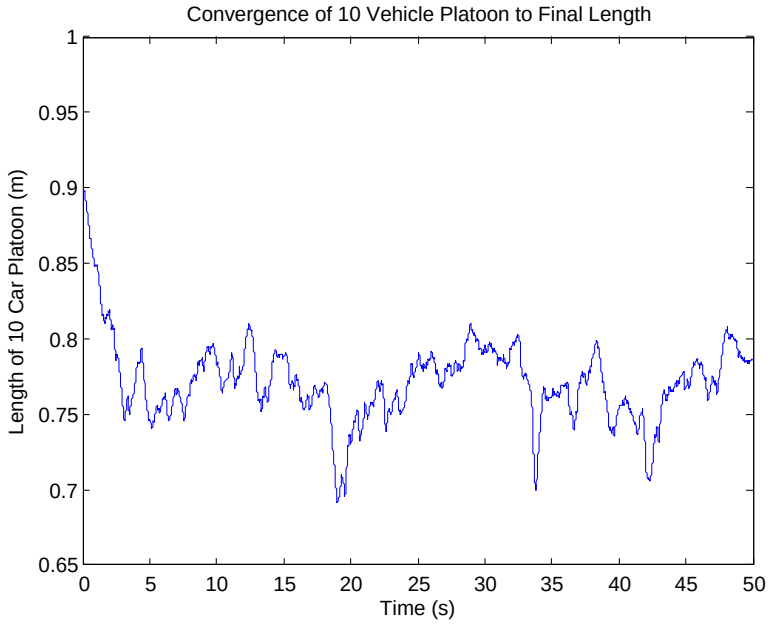


Fig. 4.16: Low frequency oscillations in the platoon appear when kinematic variation is introduced, causing the overall platoon length to decrease.

the oscillations are due to the impulses caused by the changes of the acceleration at discrete times. Varying the damping ratio, ξ, in the control algorithm, Eq. (4.1), did not eliminate the oscillations so the control law itself was modified.

4.3 Modified Control Algorithm

The control algorithm described by Eq. (4.1) is essentially a modified spring system. Simulation results demonstrated the effectiveness of this design when kinematic data was precise. When variation is introduced into the system, however, the overall platoon length began shrinking. This can be rationalized by noting Eq. (4.1) does not account for the distance between a following vehicle and the lead vehicle. Hence, the control algorithm cannot directly determine nor correct a vehicle's position with respect to the leader.

Given the relative symmetry of Eq. (4.1) for velocity and acceleration, it is natural to extend the balance between lead and preceding vehicle data into a new position term for the lead vehicle. The modified control algorithm then becomes:

$$\ddot{x}_i^* = (1 - C_1)\ddot{x}_{i-1} + C_1\ddot{x}_0 - (2\xi - C_1(\xi + \sqrt{\xi^2 - 1}))\omega_n\dot{\varepsilon}_i \tag{4.3}$$

$$-(\xi + \sqrt{\xi^2 - 1})\omega_n C_1(\dot{x}_i - \dot{x}_0) - \{(1 - C_1)\varepsilon - KC_1(x_0 - iL - x_i)\}\omega_n^2,$$

where K is a tuning variable and the product iL provides the ideal spacing between Vehicle i and the lead vehicle where L is the ideal_gap. All shared information in Eq. (4.3) is still contained in the envisioned data packet of Figure 2.3.

For a 10-vehicle platoon traveling at a constant velocity of 20 m/s, experimental evidence showed a value of K=4.0 consistently maintained vehicle spacing with increasing kinematic variation. Figure 4.17 compares the platoon length when vehicle acceleration is governed by the original controller of Eq. (4.1) to the length when the modified controller from Eq. (4.3) is implemented for increasing standard deviations. Incorporating the spacing error between a vehicle and the lead vehicle better maintains the overall platoon length when vehicles are exposed to larger kinematic variation than the original controller. For a standard deviation of magnitude 0.04, the original controller resulted in a platoon length

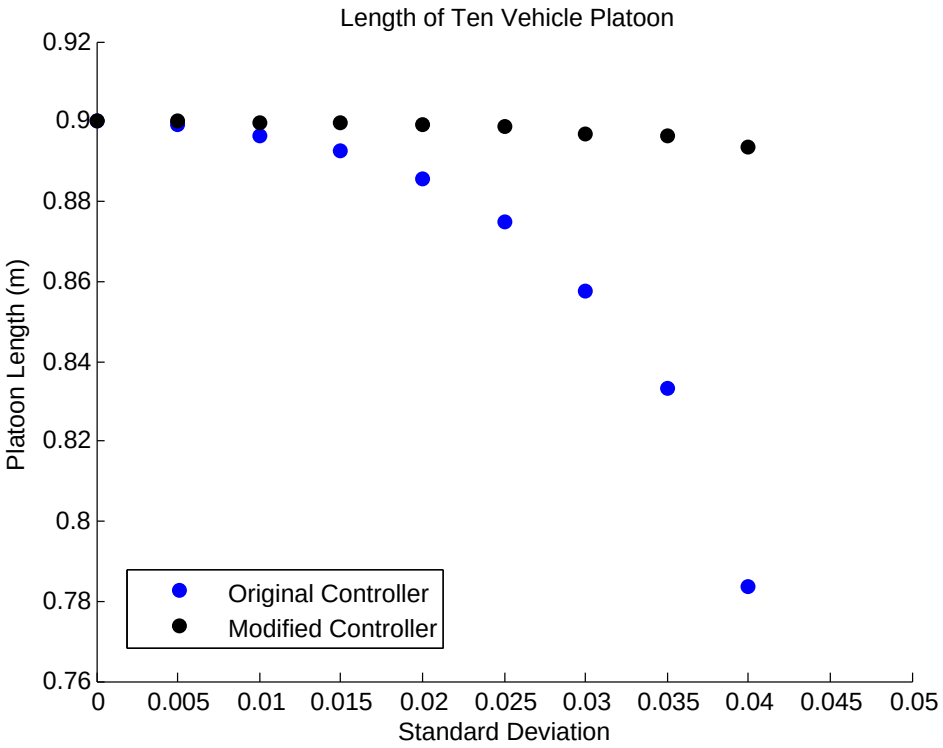
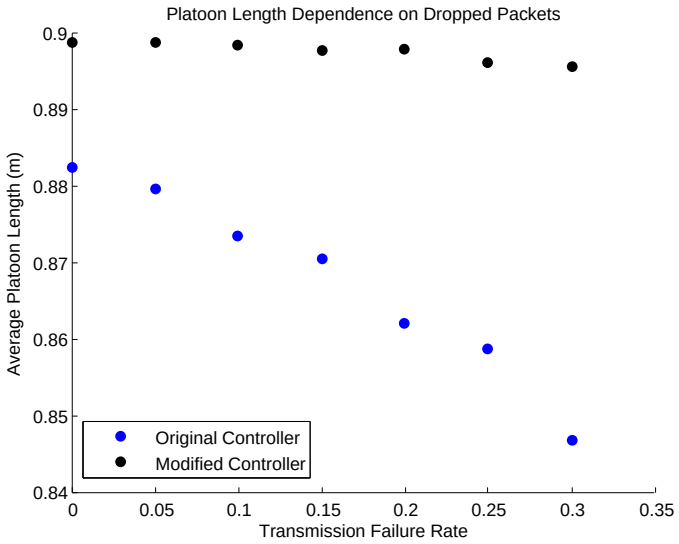


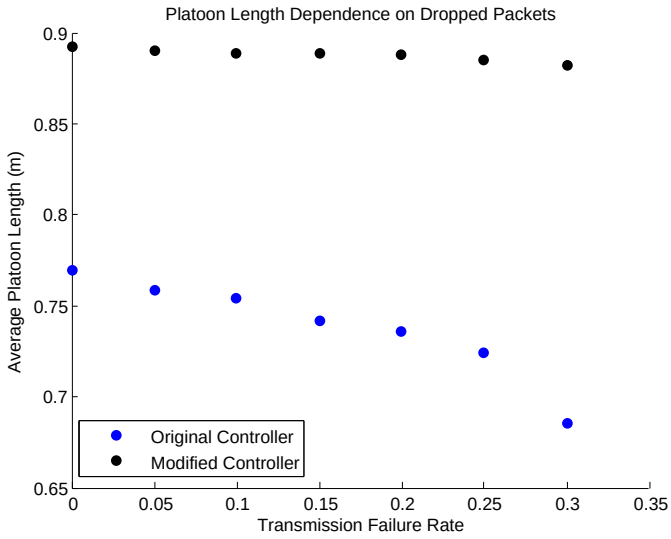
Fig. 4.17: The modified control algorithm maintains the platoon length better than the control algorithm of Eq. (4.1) in the presence of kinematic variation.

of 0.7835 m which is over one ideal_gap smaller than the ideal platoon length of 0.9 m. The modified controller was more resilient to the increased variation with an overall platoon length of 0.8937 m for the same simulation.

The modified controller, Eq. (4.3), was developed to stabilize the platoon length. As the platoon will suffer from dropped transmission packets, it needs to be tested under those conditions. Simulating a ten vehicle platoon with initial vehicle spacing equal to the ideal_gap of 0.1 m and moving at a constant speed of 20 m/s, the differences between the original controller, Eq. (4.1), and the modified controller are seen in Figure 4.18(a) and Figure 4.18(b) for variations with standard deviation of magnitude 0.02 and 0.04, respectively. With a standard deviation of 0.04 in the transmitted data and a transmission failure rate of 30%, the original controller causes the platoon to shrink by 24%. With the same standard deviation and failure rate, the modified controller results in a platoon



(a) Kinematic variation with standard deviation of 0.02.



(b) Kinematic variation with standard deviation of 0.04.

Fig. 4.18: Average length of ten vehicle platoon for the controller of Eq. (4.1) and the modified controller in Eq. (4.3). The modified controller maintains the overall platoon length noticeably better than the original controller, even in the presence of unsuccessful transmissions.

shrinkage of 2%. Unfortunately, in the present form the modified controller does not reduce the wasted energy within the platoon. The modified controller, after a time period of 50 seconds, resulted in a relative energy loss of 14252 J/kg. This value compares to the relative energy loss of 14351 J/kg for the original controller.

For the standard deviation in transmitted kinematic data of 0.04 with a failure rate in transmission of 30%, both controllers generated accelerations which exceeded the physical bounds of ± 3 m/s^2 causing railing. The modified controller was better at maintaining the overall platoon length. The relative energies, however, show the inter-vehicle gaps were varying about equally in both cases.

4.4 New Controller with Adaptive Feature

The modified controller is not making use of all the relevant information available to it. C_1 is a measure of each vehicle's confidence in the lead vehicles information compared to the preceding vehicle: C_1=1 implies Vehicle i is relying solely on data from the lead vehicle while C_1=0 implies total reliance on the preceding vehicle. For the simulations ran prior to this point, C_1 was 0.5, giving equal weight to the two sets of transmitted data. However, all the trailing vehicles know which vehicles were successful in their transmissions of kinematic data. This information will allow each trailing vehicle to adapt its control algorithm to the appropriate situation by scaling C_1. If the lead vehicle is unsuccessful in transmitting its data, all platoon members should change C_1 to 0, eliminating any use of the lead vehicles information for that period. If Vehicle i-1 fails to complete a transmission of its kinematic data, Vehicle i should change its C_1 to 1 unless the lead vehicle also failed in its transmission. In this case, Vehicle i should not update its acceleration but instead continue operating under its old acceleration and velocity values.

When simulated at a constant velocity of 20 m/s, the ten vehicle platoon, using this adaptive control algorithm, produced essentially identical results to the modified system. The standard deviation magnitude in transmitted data was varied between 0.01 and 0.04. The failure rate was varied between 0% and 30%.

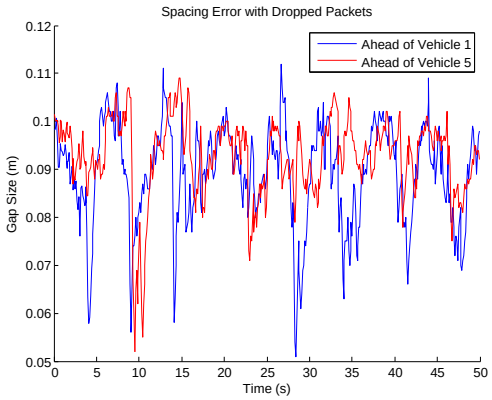
When the inter-vehicle gap was examined, however, a significant change was noted.

Figure 4.19(a) shows the relative spacing between the lead vehicle and Vehicle 1 as well as the relative spacing between Vehicle 4 and Vehicle 5 in a 10 vehicle platoon moving according to the velocity trajectory described by Eq. (4.2) with γ=1.0 when vehicles are initially spaced at the ideal_gap. The failure rate in transmissions was 25% with a standard deviation magnitude of 0.035. The controller for the simulation was the original PATH-demonstrated controller of Eq. (4.1). Figure 4.19(b) uses the controller in Eq. (4.3) with the same conditions. Finally, Figure 4.19(c) uses Eq. (4.3) in combination with the adaptive C_1 which switches between {0, 0.5, 1.0} based on transmission success. All simulations were ran for 50 seconds with a time step of 10 ms and a reaction delay of 1 ms. Note, the oscillations due to the original controller are much larger in amplitude and the gap size reflects more frequencies. The larger deviations from the ideal spacing are towards the smaller gap spacing, resulting in a decrease in the platoon length.

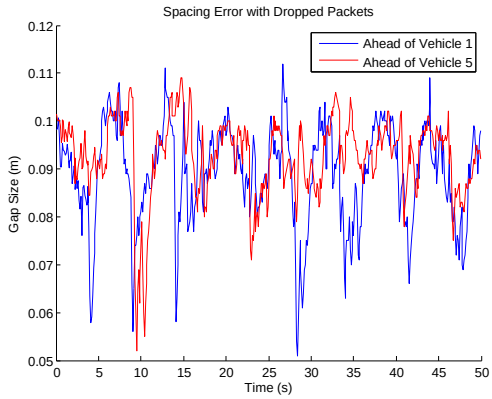
Contrasting the difference between the adaptive controller and the modified controller, Eq. (4.3) is difficult on these graphs. A direct comparison of the gap between Vehicles 4 and 5 for the two controllers is shown in Figure 4.20. The adaptive controller has a slightly smaller oscillation in its amplitude and lacks the large deviation from the ideal gap size shown by the modified controller, with the spike 47.7 seconds into the simulation being an exception. Both controllers produce an oscillation in the system, one which might be controlled via signal processing.

4.5 Signal Processing

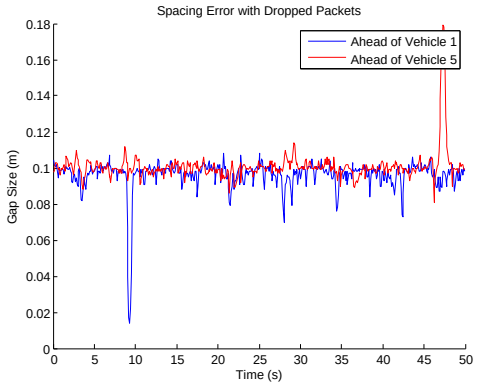
Although implementing the modified controller of Eq. (4.3) reduced the impact of kinematic variation on platoon length, the new controller did not alter the energy expenditure. As briefly mentioned earlier, even the small variation introduced into the velocity and acceleration values caused a significant increase in energy expenditures. The values of σ implemented do not correspond to a large variation in speed. Choosing σ=0.01 m/s translates into 67% of the encoded velocity values being between 26.99 m/s and 27.01 m/s for an average velocity of 27 m/s. Even bumps on the road can produce larger variations in velocity. Hence, kinematic variation will occur in the physical implementation.



(a) Vehicle desired accelerations derived from Eq. (4.1).



(b) Vehicle desired accelerations derived from Eq. (4.3).



(c) Vehicle desired accelerations derived from Eq. (4.3) but with C_1 scaled adaptively.

Fig. 4.19: Relative vehicle spacing between lead vehicle and Vehicle 1 as well as between Vehicle 4 and Vehicle 5 when tracking Eq. (4.2) with γ=1.0 assuming a 25% transmission loss rate and kinematic variation with standard deviation 0.035.

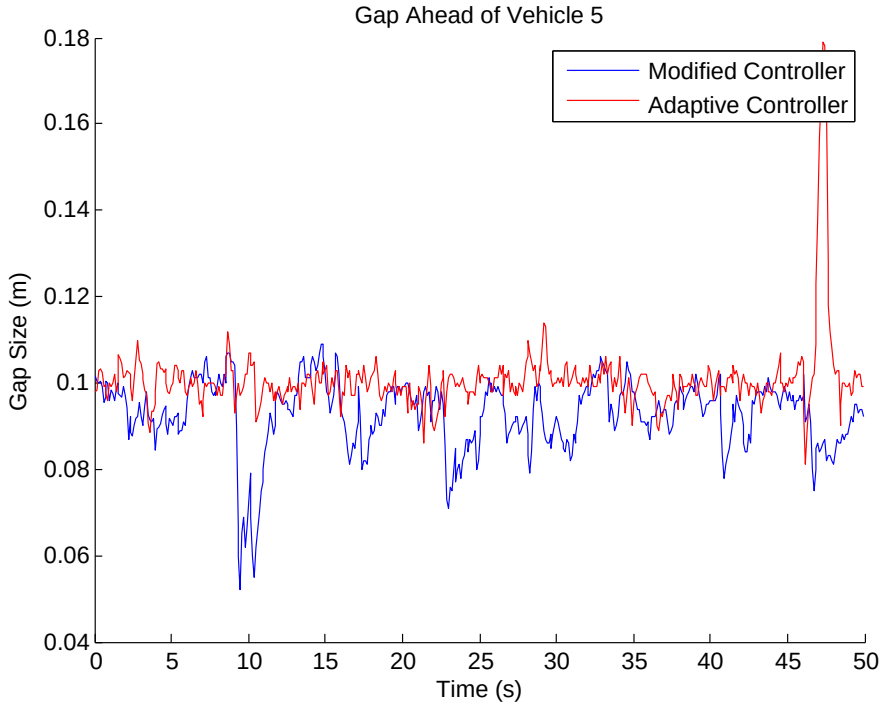


Fig. 4.20: A direct comparison between the modified controller of Eq. (4.3) and the adaptive controller: relative spacing between Vehicles 4 and 5 while following a velocity trajectory of Eq. (4.2) with γ=1.0, a 25% transmission loss rate, and variation with standard deviation of magnitude 0.035.

Signal processing algorithms can be implemented to reduce kinematic variation and improve vehicle efficiency. To explore the impact of signal processing on the large energy expenditures accrued in the kinematic variation scenarios, a simple least-squares technique was added to the code. Each vehicle would store its previous ten accelerations into an array called Quadratic. A quadratic least-squares fit was applied to the ten acceleration values to produce a predicted acceleration. This predicted acceleration, $\hat{\ddot{x}}$, for Vehicle i was combined with the controller produced acceleration, $\ddot{x}^*$ according to:

$$\ddot{x}_i^* = R\ddot{x}_i^* + (1-R)\hat{\ddot{x}}_i, \tag{4.4}$$

where R is an experimentally determined weighting factor. Experimentation showed R=0.8 yielded the best results in term of energy savings. Equation (4.4) has the additional effect

of adding inertia to the system as it slows the acceleration of vehicles. Although a relatively simple signal processing technique, this approach noticeably reduced the abrupt velocity changes resulting from kinematic variations. The corresponding decrease in wasted energy is presented in Figure 4.21 when ten vehicles traveled at a constant velocity of 20 m/s.

Similar to the control system, the high speed of vehicles within a platoon requires fast signal processing algorithms. Using a quadratic least-squares fit is a comparatively primitive technique but is fast and shows the benefits of introducing signal processing to the overall platoon system. Sophisticated, high-speed signal processing algorithms exist. Though implementing a more advanced processing scheme would most likely further reduce the energy consumption of following vehicles subject to kinematic variation and potentially reduce the inter-vehicle oscillations, a more sophisticated signal processing algorithm exceeds the scope of this work.

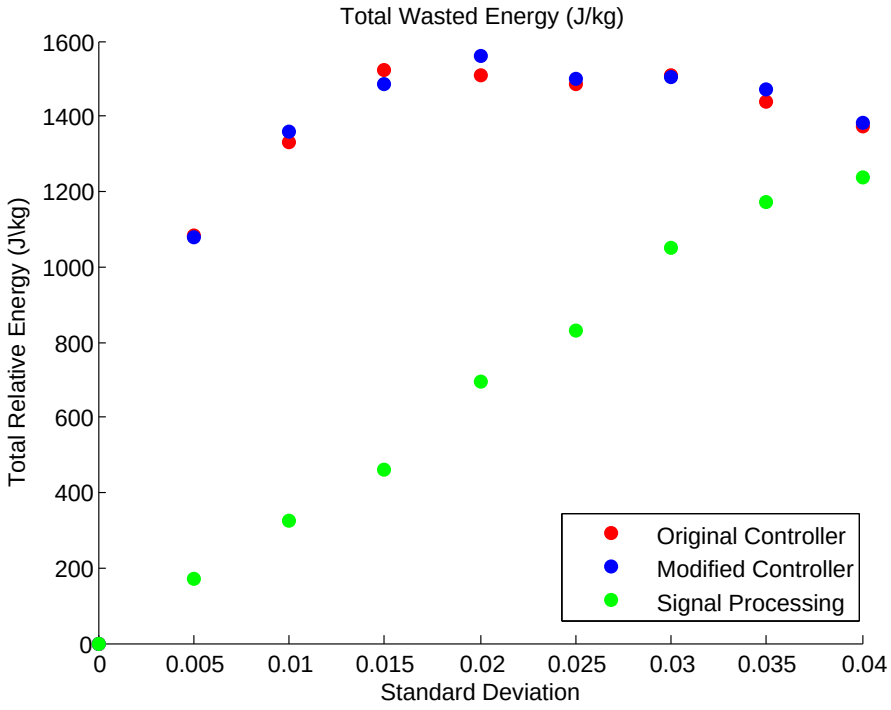


Fig. 4.21: Signal processing can counter kinematic variation and improve platoon efficiency. Here, a simple quadratic fitting algorithm was implemented.

4.6 Chapter Summary

Although successfully implemented in the eight-vehicle PATH demonstration, the performance of the control algorithm developed by Rajamani et al. [2], Eq. (4.1), deteriorated significantly in the presence of kinematic variation. Oscillations in spacing and velocity errors appeared with greater frequency and magnitude as the standard deviation of variation was increased. Large increases in relative energy expenditures accompanied the oscillations.

The oscillatory behavior was further exacerbated by increasing transmission loss rate. Although relative vehicle spacings oscillated around the ideal_gap, the presence of kinematic variation caused the oscillations to tend toward smaller vehicle spacings. This collective behavior resulted in the platoon length shrinking during the course of each simulation.

Modifying the original control algorithm of Eq. (4.1) to include the distance between Vehicle i and the lead vehicle, as expressed in Eq. (4.3), significantly reduced the observed platoon shrinkage. When 30% of transmissions were unsuccessful and the kinematic variation had standard deviation of 0.04, the original controller causes the platoon to shrink by 24%. With the same standard deviation and failure rate, the modified controller results in a platoon shrinkage of just 2%.

Because vehicles are aware of which platoon members successfully transmit, the modified controller can be further improved by adaptively tuning C_1 in Eq. (4.3). C_1 is essentially a ratio of the confidence Vehicle i assigns to the lead vehicle with respect to the preceding vehicle. When the preceding vehicle fails to successfully transmit, a vehicle adjusts C_1 to 1.0 and relies entirely on data transmitted by the lead vehicle. Alternatively, if the lead vehicle's transmission is unsuccessful, C_1 is equal to 0 so only the preceding vehicle's data is used in Eq. (4.3). When both vehicles transmit successfully, C_1 is 0.5 as in early simulation work. If both the lead vehicle and preceding vehicle are unsuccessful, Vehicle i maintains its velocity and acceleration for the next period. The adaptive control scheme was comparable to the modified controller at maintaining overall platoon length. Individual vehicles benefited from slightly smaller oscillations about the ideal_gap.

Neither the modified controller nor the adaptive controller reduced the high relative energy expenditures encountered with increasing variation. Instead, a quadratic least-squares approximation was used to generate a predicted acceleration which was then combined with the controller-generated acceleration. The result of this simple signal processing technique was a more efficient platoon. When vehicles traveled at a constant 20 m/s, a standard deviation of just 0.005 caused the ten-vehicle platoon to expend over 1000 J/kg during the 50 second simulation run prior to the signal processing. Using the quadratic least-squares, the same platoon required less than 200 J/kg.

Overall, the development and results of three novel ideas were presented in this chapter: modified controller, adaptable confidence ratio C_1, signal processing to reduce kinematic variation. Combining these ideas improved vehicle behavior, reduced platoon shrinkage, and decreased relative energy expenditures.

Chapter 5

Summary of Results and Future Work

Vehicle plaooning has been successfully demonstrated in controlled environments. The 1997 eight-vehicle PATH demonstration took place on an isolated track. Radio communication relayed information between vehicles for use in the longitudinal controller which maintained a 6.5 m vehicle gap, with errors of ± 0.2 m. The velocity trajectory for the demonstration began at rest, increased to 60 mph (27.9 m/s), then returned to rest in a 100 s window [1, 2]. A European group, SARTRE, has similarly implemented a physical platoon. SARTRE vehicles were spaced as closely as 4 m apart during a 200 km test along actual highways in Spain [7]. Very little technical information has been disclosed regarding SARTRE's control algorithm though wireless communication was used to share information between vehicles.

SARTRE was able to further reduce vehicle spacing to slightly improve fuel consumption. Balancing improved efficiency and safety concerns indicates a vehicle gap of 0.1 m will significantly reduce vehicle fuel consumption without increasing collision damage. This small vehicle spacing requires high-speed communication and robust control. Within this thesis, the effectiveness of PATH's well-documented control algorithm, Eq. (3.3), was evaluated when vehicles communicate per the WAVE protocol to maintain a vehicle spacing of 0.1 m.

To analyze Eq. (3.3), a computer simulation was developed in C++ to model platoon behavior. The lead vehicle received a user-defined velocity trajectory with acceleration determined from the velocity using numerical integration. Each following vehicle attempted to remain 0.1 m behind its predecessor while following the lead vehicle. The desired acceleration for each following vehicle was determined by the control algorithm. When modeling Eq. (3.3), vehicles use the current velocity and acceleration of the preceding and lead vehicle

to determine a desired acceleration to be implemented in the next period.

PATH relied on radio communication to share this information but wireless communication has emerged as the preferred protocol for vehicular networks. In the United States, WAVE was developed around the established IEEE 802.11 family of standards [22]. The physical layer of WAVE is governed by 802.11p which essentially doubles the data rate of 802.11a but does not alter the header or preamble structure of data packets.

Under the recently accepted IEEE 802.11p amendment, as part of WAVE, a model was developed for this work which found a platoon of 25 vehicles could transmit position, velocity, and acceleration data every 10 ms with less than a 20% loss rate. As a result of this analysis, a period of 10 ms was chosen for the control simulation as well. The validity of the communication model was supported by adjusting the data transmission rates to match those specified in IEEE 802.11a and looking at the channel throughput. Although the simulation increased the payload size by adding additional packets, hence increasing the unusable bits which varies from standard throughput analysis, the simulated throughput was found to have the same functional form as documented throughput rates for IEEE 802.11a.

The control algorithm was similarly validated by comparing simulated results to documented behavior. To model the results of PATH as presented by Rajamani, the simulation assumed all transmissions were successful so every vehicle had the data necessary to calculate its desired acceleration for the next period. The implemented acceleration was bounded by ± 3 m/s^2 to model physical limitations of actual vehicles. With this ideal communication, vehicles were able to maintain the 0.1 m gap when the lead vehicle tracked four separate trajectories, which increased in frequency, of the functional form:

$$\ddot{x}_i = 20.0 + sin(t/\gamma), \tag{5.1}$$

where i={1,2,...,N} and N is the number of platoon members. The velocity trajectories explored in the simulation are more severe than those implemented by PATH to ensure robust control.

The original controller also maintained vehicle spacing when kinematic data remained exact but the transmission loss rate was increased to 20%, where one in five transmissions were unsuccessful. Larger velocity impulses were needed to maintain ideal spacing however, which translated to larger energy expenditure. When 30% of transmissions were unsuccessful, the spacing errors began amplifying with time making this loss rate unacceptable for physical implementations.

Due to sensor accuracy and environmental conditions, a physical platoon will also encounter kinematic variation. The control algorithm will therefore need to accommodate inaccurate velocity and acceleration values. To isolate the influence of variation, the velocity trajectory of the lead vehicle was reduced from a sinusoid to a constant velocity of 20 m/s. Returning to the ideal communication scenario but introducing variation with a standard deviation of 0.02 in the kinematic data noticeably reduced the ability of Eq. (3.3) to maintain vehicle spacing. Relatively large oscillations occurred in both the spacing error between vehicles and the relative velocity with transmission losses aggravating this behavior. Velocity oscillations caused a large increase in the relative energy required by vehicles which is discouraging. Even more troublesome, the overall platoon length actually decreased in the presence of kinematic variation.

Rajamani observed oscillations in the PATH implementation as well. With a vehicle spacing of 6.5 m, the spacing error of 0.2 m was not a concern. Reducing the vehicle gap to 0.1 m makes this spacing error unacceptable, however. A new controller design is needed which will better maintain the platoon length when transmitted position, velocity, and acceleration data have some deviation from the exact values. Building on the symmetry of Eq. (3.3), an additional position term was added to the control algorithm, resulting in Eq. (4.3). The new term is a measure of the spacing between Vehicle i and the lead vehicle.

The modified control algorithm of Eq. (4.3) was significantly more effective at maintaining overall platoon length. When 30% of transmissions were unsuccessful and the kinematic variation had standard deviation of 0.04, the original controller causes the platoon to shrink by 24% when vehicles follow the velocity trajectory of Eq. (5.1) with γ=1. With the same

standard deviation and failure rate, the modified controller results in a platoon shrinkage of just 2%.

Individual vehicles still experienced frequent oscillations in position and velocity under the modified controller. In an effort to reduce this internal oscillation, C_1 in Eq. (4.3) was changed from a fixed constant of 0.5 to an adaptive gain. Because C_1 reflects the confidence of Vehicle i in the lead vehicle's data with respect to the preceding vehicle's data, the value of C_1 should depend on which vehicles successfully transmitted. The adaptive scheme did reduce internal oscillations slightly but the overall energy expenditure for vehicles exposed to kinematic variation was still much larger than the ideal scenario.

Signal processing can reduce the impact of kinematic variation on platoon efficiency. As a demonstration, a simple quadratic least-squares algorithm was implemented. Predicting the next acceleration based on the previous 10 acceleration values and combining with the controller-generated acceleration significantly decreased internal oscillations within the platoon. When vehicles traveled at a constant 20 m/s, a standard deviation of just 0.005 in the kinematics caused the ten-vehicle platoon to expend over 1000 J/kg during the 50 second simulation run prior to the signal processing. Using the quadratic least-squares as a demonstration, the same platoon required less than 200 J/kg.

A more advanced signal processing algorithm could potentially further reduce the energy expenditure by removing vehicular oscillations and would be worth investigating. The simulations conducted in this work were compared to published works as a means of validation, but the results are still simulated. Ultimately, it would be extremely valuable to implement the modified control algorithm in physical vehicles to get experimental data. On the communication end, WAVE has multiple channels allotted for vehicular communication. This work assumed a single communication channel but it would be interesting to see if a multi-channel network could better accommodate multiple platoons. Finally, only longitudinal control was considered in this thesis. A very similar analysis could be undertaken for lateral control with the two potentially being merged to provide a model for more extensive platooning scenarios.

More work is needed before full-scale implementation of vehicular platoons is feasible but the modified controller in Eq. (4.3) with adaptable confidence gain C_1 shows significant promise. It is more robust to transmission losses and especially kinematic variation than the already implemented controller of Eq. (3.3). Including a signal processing algorithm reduces the negative affects of kinematic variation, thus improving platoon efficiency which provides financial and environmental incentives for implementing platooning systems.

References

[1] California Partners for Advanced Transit and Highways. (1997, November) Vehicle platooning and automated highways. [Online]. Available: http://www.path.berkeley.edu/PATH/Publications/Media/FactSheet/index.html

[2] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang, "Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 695–708, July 2000.

[3] X. Liu, A. Goldsmith, S. Mahal, and J. Hedrick, "Effects of communication delay on string stability in vehicle platoons," *Intelligent Transportation Systems. IEEE Proceedings*, pp. 625–630, 2001.

[4] S. Grafling, P. Mahonen, and J. Riihijarvi, "Performance evaluation of IEEE 1609 WAVE and IEEE 802.11p for vehicular communications," *Ubiquitous and Future Networks (ICUFN). Second International Conference*, pp. 344–348, June 2010.

[5] S. S. Jackson, "Safety aware platooning of automated electric transport vehicles," Master's thesis, Utah State University, Logan, UT, 2012.

[6] Texas A&M Transportation Institute. (2013, February) 2012 annual urban mobility report. [Online]. Available: http://mobility.tamu.edu/ums/media-information/press-release/

[7] Safe Road Trains for the Environment. (2012, September) SARTRE final partner release pdf. [Online]. Available: http://www.sartre-project.eu/en/press/Sidor/default.aspx

[8] California Partners for Advanced Transit Technology. [Online]. Available: www.path.berkeley.edu

[9] S. Shladover, C. Desoer, J. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, "Automated vehicle control developments in the PATH program," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, Feb 1991.

[10] IEEE Standards Association. IEEE standard 802.11p-2010. [Online]. Available: http://standards.ieee.org/findstds/standard/802.11p-2010.html

[11] R. Rajamani, *Vehicle Dynamics and Control*. New York: Springer-Verlag, 2006.

[12] P. Fernandes and U. Nunes, "Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 91–106, Mar. 2012.

[13] A. Bohm and M. Jonsson, "Position-based data traffic prioritization in safety-critical, real-time vehicle-to-infrastructure communication," *Communications Workshops, 2009 IEEE International Conference*, pp. 1–6, June 2009.

[14] P. Fernandes and U. Nunes, "Platooning of autonomous vehicles with intervehicle communications in SUMO traffic simulator," *Intelligent Transportation Systems. 13th International IEEE Conference*, pp. 1313–1318, Sept. 2010.

[15] S.-Y. Pyun, D.-H. Cho, and J.-W. Son, "Downlink resource allocation scheme for smart antenna based V2V2I communication system," *IEEE Vehicular Technology Conference*, pp. 1–6, Sept. 2011.

[16] T. Tank and J.-P. Linnartz, "Vehicle-to-vehicle communications for AVCS platooning," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 528–536, May 1997.

[17] T. Sukuvaara and P. Nurmi, "Wireless traffic service platform for combined vehicle-to-vehicle and vehicle-to-infrastructure communications," *IEEE Wireless Communications*, vol. 16, no. 6, pp. 54–61, Dec. 2009.

[18] D. Khadraoui and T. Sukuvaara, "Wireless traffic service communication platform for cars," *Global Information Infrastructure Symposium. GIIS '09*, pp. 1–7, June 2009.

[19] Y. Xiao and J. Rosdahl, "Throughput and delay limits of IEEE 802.11," *IEEE Communications Letters*, vol. 6, no. 8, pp. 355–357, Aug. 2002.

[20] Microsoft. (2003) How 802.11 wireless works. [Online]. Available: http://technet.microsoft.com/en-us/library/cc757419(v=ws.10).aspx

[21] (2004) Chapter 8: 802.11 PHY layers. [Online]. Available: http://cdn.ttgtmedia.com/searchMobileComputing/downloads/CWAP_ch8.pdf

[22] R. Uzcategui and G. Acosta-Marum, "WAVE: A tutorial," *IEEE Communications Magazine*, vol. 47, no. 5, pp. 126–133, May 2009.

[23] I. Ivan, P. Besnier, M. Crussiere, M. Drissi, L. Le Danvic, M. Huard, and E. Lardjane, "Physical layer performance analysis of V2V communications in high velocity context," *Intelligent Transport Systems Telecommunications. 9th International Conference*, pp. 409–414, Oct. 2009.

[24] W. Vandenberghe, I. Moerman, and P. Demeester, "Approximation of the IEEE 802.11p standard using commercial off-the-shelf IEEE 802.11a hardware," *ITS Telecommunications. 11th International Conference*, pp. 21–26, Aug. 2011.

[25] M. Aumann. (2011, April) Inside NASCAR: The science of the tandem draft. [Online]. Available: http://www.nascar.com/en_us/news-media/articles/2011/04/20/inside-nascar-science-draft.htm

[26] A. Hitchcock. (2013, February) Intelligent vehicle/highway system safety: Multiple collisions in automated highway systems. Partners for Advanced Transport and Highways. [Online]. Available: http://www.escholarship.org/uc/item/6vm8z32v

[27] National Highway Traffic Safety Administration. (1990, April) Bumper questions and answers. [Online]. Available: http://www.nhtsa.gov/cars/problems/studies/bumper/index.html

[28] A. Pue, "Implementation trade-offs for a short-headway vehicle-follower automated transit system," *IEEE Transactions on Vehicular Technology*, vol. 28, no. 1, pp. 46–55, Feb. 1979.

[29] D. McMahon, J. Hedrick, and S. Shladover, "Vehicle modelling and control for automated highway systems," *American Control Conference*, pp. 297–303, May 1990.

[30] Y. Bin, K. Li, X. Lian, H. Ukawa, M. Handa, and H. Idonuma, "Longitudinal acceleration tracking control of vehicular stop-and-go cruise control system," *IEEE International Conference on Networking, Sensing and Control*, vol. 1, pp. 607–612, March 2004.

[31] D. Olson and W. L. Garrard, "Model-follower longitudinal control for automated guideway transit vehicles," *IEEE Transactions on Vehicular Technology*, vol. 28, no. 1, pp. 36–45, Feb. 1979.

[32] L. Nouveliere and S. Mammar, "Experimental vehicle longitudinal control using second order sliding modes," *American Control Conference. Proceedings*, vol. 6, pp. 4705–4710, June 2003.

[33] T. S. No, K.-T. Chong, and D.-H. Roh, "A lyapunov function approach to longitudinal control of vehicles in a platoon," *IEEE 51st Vehicular Technology Conference. VTC 2000*, vol. 1, pp. 336–340, 2000.