5-2023

VANET Applications Under Loss Scenarios & Evolving Wireless Technology

Adil Alsuhaim
aalsuha@clemson.edu

VANET Applications under Loss Scenarios & Evolving Wireless Technology

---

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Computer Science

---

by
Adil Abdulaziz S. Alsuhaim
May 2023

Accepted by:
Dr. Long Cheng, Committee Chair
Dr. Abolfazl Razi
Dr. Amy Apon
Dr. Jim Martin
Dr. James Westall

Abstract

In this work we study the impact of wireless network impairment on the performance of VANET applications such as Cooperative Adaptive Cruise Control (CACC), and other VANET applications that periodically broadcast messages. We also study the future of VANET application in light of the evolution of radio access technologies (RAT) that are used to exchange messages. Previous work in the literature proposed fallback strategies that utilizes on-board sensors to recover in case of wireless network impairment, those methods assume a fixed time headway value, and do not achieve string stability. In this work, we study the string stability of a one-vehicle look-ahead CACC platoon under different network loss scenarios, and propose to adapt the time headway parameter of the model according to a network reliability metric that we defined based on packet burst loss length to maximize traffic flow efficiency while maintaining a string stable platoon. Our findings show that careful adjustment of headway value according to the wireless network reliability allows the platoon to maintain string stable operation while maximizing traffic flow. We also study the impact that evolving wireless technology can have on VANET applications such as CACC, where we study the performance when using DSRC and 5G NR V2X.

In addition, we study the evolution of RATs used in VANET application, and we propose DSRC+, as a possible enhancement to traditional DSRC, that utilizes modern modulation/coding schemes and performs random blind retransmission to improve packet delivery ratio. We finally study the trade-offs in the choice of RAT in VANET applications such as CACC, concluding that RATs with time-division channel access can be reliable with lower packet loss, but performs poorly when needing to disseminate messages over longer CACC platoons.

Dedication

Dedicated to the Kava Konnection Family in Greenville, South Carolina.

Acknowledgments

I would like to thank Dr. Jim Martin for his valuable time, and continuous support throughout this work. I would also like to thank Dr. James M. Westall for his continuous support, encouragement, and helping me get my paper accepted in IEEE Transactions on Vehicular Technology. I could not have done this without your help.

I would also like to thank Dr. Long Cheng, and Dr. Abolfazl Razi for their continuous help, and conducting weekly Zoom meeting to discuss the progress of this work.

I would like to thank Dr. Heather Brannon, MD, of Greenville ADHD Specialists for providing me with tremendous guidance to my mental health that positively impacted my social & school life.

Clemson University is acknowledged for generous allotment of compute time on Palmetto cluster. The Palmetto cluster was very essential in helping me run many simulation experiments in parallel, and collecting the results in a relatively short time.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Recent technological developments in wireless technologies lead to the advent of Vehicular Ad hoc NETworks (VANETs). VANETs are a special type of Mobile Ad hoc NETworks (MANETs), in which network nodes are vehicles equipped with wireless communication devices. Wireless Access in Vehicular Environment (WAVE) was specified by IEEE 802.11p in 2010 to add Dedicated Short-Range Communication (DSRC) between vehicles in the 5.9 GHz band. It reserves a 75 MHz band consisting of seven 10 MHz channels for use in vehicular communication.

VANETs have attracted significant research interest from academicians and corporations, with a particular focus on Cooperative Adaptive Cruise Control (CACC) applications due to its potential in improving traffic flow, and therefore reduction in traffic congestion, fuel costs and emissions. Automobile manufacturers have been recently offering drivers the ability to engage an Adaptive Cruise Control (ACC) mode. In a traditional cruise control driving, a driver sets a desired speed, and the vehicle maintains that speed. In a vehicle equipped with ACC, the driver would set a desired cruise speed, and the vehicle attempts to maintain that speed, but uses on-board sensors to estimate distance to the preceding vehicle, and reduces speed to avoid a collision. The availability of DSRC in vehicles allows them to share information, such as acceleration, with one another. CACC leverages information shared by vehicles to achieve more optimal traffic flow by forming a platoon of

vehicles on the road with smaller separation margins. CACC is particularly interesting to transport companies because it can reduce the operating costs of trucks on the highways.

DSRC-equipped vehicles do not typically have a centralized access coordinator. The IEEE802.11 standard[2] specifies the process by which channel access is regulated. The standard specifies a contention-based medium access control procedure called Carrier-Sense Multiple Access with Collision Avoidance (CSMA-CA). Since this process is somewhat random, it is possible that two vehicles may attempt to transmit a frame at the same time, resulting in loss of data due to frame collision. This channel access mechanism offers no performance guarantees, or bounded delays. Vehicular applications such as CACC involve vehicles broadcasting periodic beacon messages with relevant information, and broadcast frames do not get acknowledgment of reception in order to minimize channel congestion. When the number of vehicles in the vicinity increases, the probability of packet loss due to wireless frame collision increases. In addition, it is possible for a malicious node to perform a denial-of-service attack, rendering the channel useless. Failure of delivery of data can have adverse effect on the performance and safety of vehicular applications, and therefore, it is important to study these applications in light of network condition.

CACC applications aim to improve *traffic flow*, which can be defined as the number of vehicles passing through a point on the road per unit of time. A CACC application can be parameterized with a desired time headway value between vehicles. A higher headway value could mean safer operation at the expense of improved traffic flow. We would like to have this value as low as possible, while maintaining a safe and stable operation.

In addition, several emergent changes to V2X communication technologies would add uncertainty to the future of 802.11-based V2X communication. We performed simulation analysis of the performance of VANET applications such as CACC, and applications that periodically broadcast data, comparing current DSRC standard with the cellular-based 5G NR V2X sidelink communication.

In this work, we study the effect of network degradation on CACC operation, propose a metric for measuring the communication network reliability for CACC application, and use that metric to adapt the time headway parameter, h, according to that reliability metric to maximize traffic flow while maintaining string stability (defined in Section 3.1.1). Network impairment can be the result of denial-of-service attacks, total network outage, or congestion as defined in Section 3.5. The goal is to operate the platoon with the lowest h without breaking string stability. We also study the

performance of VANET applications when using the currently available radio access technologies (RATs): DSRC and 5G NR V2X, and propose DSRC+ as an enhancement to DSRC to improve the performance for VANET applications. Finally, we investigate the trade-offs of the choice of these RATs by studying how they impact the performance of CACC platoons in cases of emergency, where an emergency message needs to be disseminated to all vehicles of the platoon to indicate emergency braking. We use the time needed to notify all vehicles of that emergency, and the level of exposure to risk of collision as two metric to evaluate the two technologies.

1.2 Dissertation Outline

Chapter 2 presents an outline of the literature related to our work. We describe DSRC/WAVE in Section 2.1, with focus on the CSMA channel access mechanism, followed by a description of the Cellular-V2X channel access mechanism in 2.2. In Section 2.3 we briefly discuss safety applications in VANETs, followed by an outline of emergent technologies that would support VANETs in Section 2.4. Section 2.5 briefly outlines related work in development of CACC controllers, and a description of typical loss exhibited in VANETs.

Chapter 3 provides details of the implemented CACC controller, and we discuss the fallback strategy during network impairment in Section 3.2. We also provide a detail of simulation implementation, and present a reliability metric and time headway adaptation in Section 3.4.

Chapter 4 presents the results from our time headway adaptation algorithm. It also presents a comparison of the performance of the two wireless technologies in CACC in Section 4.2, where we compare traffic flow achievable with the minimum string stable headway when these technologies are used. In Section 4.3, we compare the packet delivery ratio (PDR) achieved with the two technologies in a scenario with stationary vehicles within proximity of each other, where these vehicles periodically broadcast messages every 100ms. Our results show that 5G NR V2X outperforms DSRC as it can achieve higher PDR in congested networks.

In Chapter 5, we implemented a Kalman filter that enhances the estimation of acceleration of a preceding vehicle in Section 5.1. In Section 5.2 we propose DSRC+, an enhancement to DSRC that utilizes modern modulation/coding schemes that would be introduced by IEEE 802.11bd, where DSRC+ employs blind retransmission to improve the performance of VANET applications.

In Chapter 6, we study the trade-offs in the choice of RAT for CACC applications, between a channel access technology that uses contention-based channel access mechanism represented by DSRC, against a technology that utilizes a mixture of time-division and carrier-sensing channel access mechanism, represented by 5G NR V2X.

Chapter 2

Literature Review

2.1 Wireless Access in Vehicular Environment (WAVE)

IEEE added Wireless Access in Vehicular Environment as an amendment to the IEEE 802.11 standard, known then as 802.11p. The 802.11p amendment has been merged with the main 802.11 standard in 2012. IEEE 802.11p reserved a 75 MHz band in the 5.9 GHz band (5.850-5.925 GHz), and hence, it is commonly referred to as "Dedicated Short-Radio Communication" (DSRC). The band is divided into seven 10 MHz channels, with channel 178 being the control channel (CCH) and the other six channels being service channels (SCH). All channels support the same modulation and coding schemes, allowing for data rates of up to 27 Mbps. IP traffic is only allowed on service channels. The standard also suggest that it is possible to create channel 175 by combining channels 174 and 176 into a 20 MHz channel, which would support modulation and coding schemes allowing for data rates up to 54 Mbps. Similarly, channels 180 and 182 can be combined into a 20 MHz channel 181 as shown in Figure 2.1. It is common to refer to the 5.9 GHz band as the Intelligent Transport Systems (ITS) band.

In Europe, the channelization is slightly different, where channel 180 is designated as CCH. The spectrum is divided into ITS-G5A, for use in safety applications, and ITS-G5B for general-purpose use as shown in Figure 2.2.

The IEEE 802.11 standard [2] specifies that the 10 MHz channels support modulation coding schemes

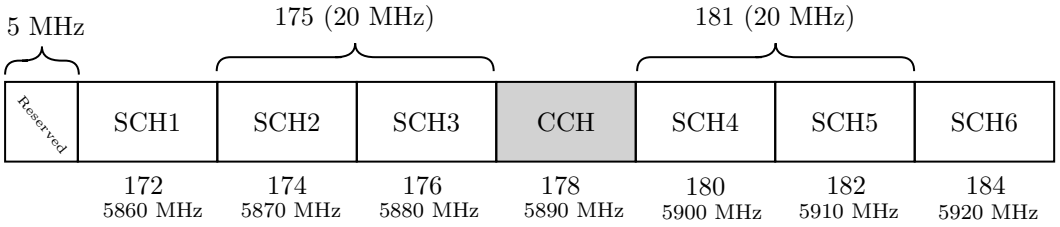


Figure 2.1: DSRC channel allocation in the United States

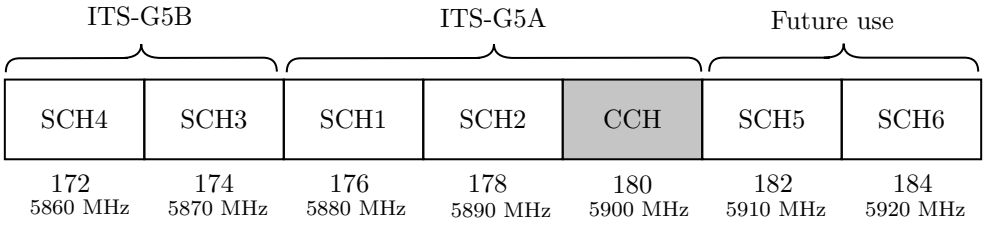


Figure 2.2: DSRC channel allocation in Europe.

supporting data rates of 3, 4.5, 6, 9, 12, 18, 24, and 27 Mbps. The 20 MHz channels would support double the data rate of the 10 MHz channels as shown in Table 2.1.

Modulation	**Coding Rate**	**Data rate (10 MHz channel) in Mbps**	**Data rate (20 MHz channel) in Mbps**
BPSK	1/2	3	6
BPSK	3/4	4.5	9
QPSK	1/2	6	12
QPSK	3/4	9	18
16-QAM	1/2	12	24
16-QAM	3/4	18	36
64-QAM	2/3	24	48
64-QAM	3/4	27	54

Table 2.1: Modulation Coding scheme and supported data rate for 10 and 20 MHz channels

A device supporting WAVE operates in a mode called Outside Context of Basic service set (OCB), which is an ad-hoc network mode that does not require the exchange of management frames for authentication and association with an access point (AP). OCB devices support all the modulation coding scheme in Table 2.1, and can choose to transmit packets at any given modulation since receivers detect the modulation from the 802.11 frame's preamble. In addition, the transmission power used can be set individually on per-frame basis to one of seven power levels between a lower bound called `TxPowerStart` and an upper end called `TxPowerEnd`. The maximum permitted transmission power is 33 dBm for non-government users such as On-Board Units (OBU), and 44.8

for government use such as in Road-Side Units (RSU). Packets transmitted at a higher modulation (i.e. with higher data rate) would require higher transmission power to travel the same distance as packets transmitted with a lower modulation.

Without the presence of an AP, devices operating in OCB mode contend for channel access using Carrier-Sense Multiple Access (CSMA) in a distributed manner. A wireless station (STA) is capable of sensing whether the wireless channel is busy or idle. STAs wanting to transmit frames would have to defer the transmission if they sensed that the wireless medium is busy. Then, if a STA senses that the medium is idle for a specific amount of time called Distributed Inter-Frame Spacing (DIFS), it generates a random backoff counter, and start counting down as long as the wireless channel remains idle. If the channel became busy during the countdown, the countdown process is stopped until it senses that the channel is idle again for a DIFS, and then it resumes counting down without generating a new backoff value. When a STA's counter reaches zero, it starts transmitting over the wireless channel. This process is illustrated in Figure 2.3. The random backoff value is in unit of slots, where each slot is 13 μs. The random value is generated as an integer between $CW_{\min}$ and $CW_{\max}$ which vary depending on the access category of the frame as shown in Table 2.2.

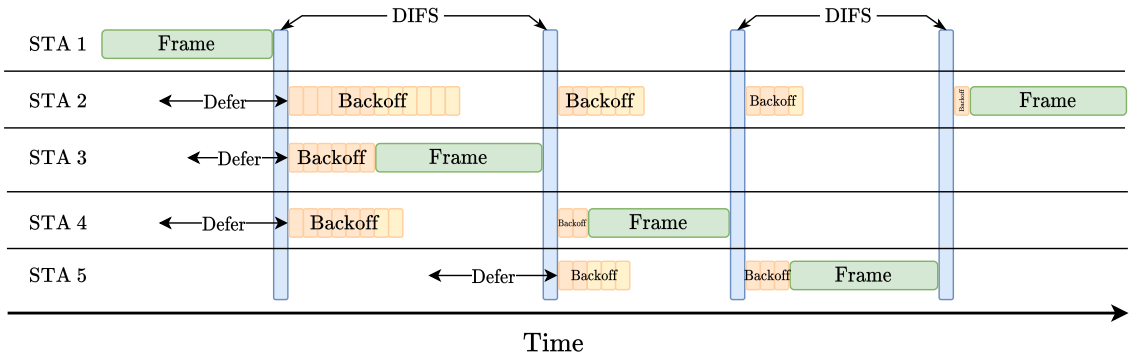


Figure 2.3: Distributed Channel Access in 802.11p

Access Category	$\boldsymbol{CW_{\min}}$	$\boldsymbol{CW_{\max}}$
Background (BK)	15	1023
Best Effort (BE)	15	1023
Video (VI)	7	15
Voice (VO)	3	7

Table 2.2: Contention Window values for different QoS access categories in 802.11

It is possible that two or more contending nodes would have their backoff timers expire at the exact time, and then attempt to transmit their frames over the wireless channel that is assumed to be

idle, causing collisions that lead to packet loss. In a congested network, this can result in excessive packet loss events, which can happen in relatively long bursts.

DSRC devices with a single physical entity (PHY) would typically tune to CCH channel in a single channel operation. Multi-channel operation [42] can be achieved by using an alternating scheme, where a 100ms synchronization interval is divided into two intervals, a CCH interval where the device is tuned to channel 178, and an SCH interval where the device is tuned to one of the SCH channels as shown in Figure 2.4. Multi-channel access can be configured to alternate between CCH and SCH every 50ms, or by giving SCH immediate access when it has data to transmit over SCH interval. Some DSRC devices are equipped with a dual-PHY radio, which can be tuned independently to any of the available channels.

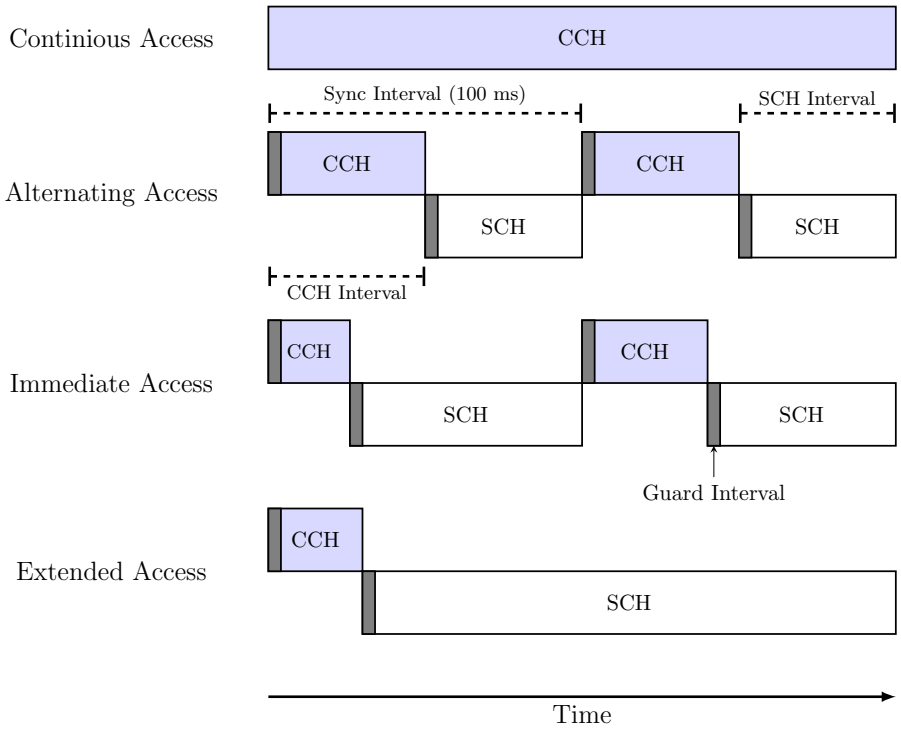


Figure 2.4: Multi-Channel Operation as defined by IEEE 1609.4

The typical Vehicular Ad-hoc Network (VANET) based on 802.11p is a flat broadcast network, where vehicles would broadcast data frames over the wireless medium to their immediate neighbors. Recipients of broadcast frames do not acknowledge their reception to reduce channel usage. The standard also supports unicast messages which requires acknowledgments.

In Japan, the band utilized for Intelligent Transport Systems is the 700 MHz band, which is specified by the Japanese Association of Radio Industries and Businesses (ARIB) in the standard "ARIB STD-T109" [65]. ARIB STD-109 uses a single 10 MHz channel centered at 760 MHz, and inherits its physical layer implementation from the 802.11 standard. However, it classifies traffic into base station (RSU) traffic and mobile traffic (vehicles), so it employs a basic time-division multiple access (TDMA) strategy where time is divided into a roadside-to-vehicle period, and inter-vehicular period that uses contention-based CSMA.

In [45], the performance of ARIB STD-T109 was compared to 802.11p, and was shown to offer increased communication range. However, although it suffered no loss in RSU traffic, it suffers a higher rate of frame collision for vehicle-to-vehicle (V2V) traffic than 802.11p due to having a smaller time window to contend for channel access.

2.2 Cellular-Vehicle-to-Everything (C-V2X)

Cellular communication is widely available in commercial deployments. The 3rd Generation Partnership Project (3GPP) standardizes cellular communication in what is known as *Releases*. User devices such as cell phones are referred to as User Equipment (UE), whereas the point of contact with the cellular network (i.e. the cellular tower) is known as E-UTRAN Node B, which is commonly shortened as eNodeB or eNB. Between a UE and an eNodeB, channels are designated for uplink and downlink traffic. For uplink, there are the Physical Uplink Control Channel (PUCCH), and Physical Uplink Shared Channel (PUSCH), and similarly for downlink we have Physical Downlink Control Channel (PDCCH) and Physical Downlink Shared Channel. Channel access can follow Frequency-Division Multiplexing (FDM), or Time-Division Multiplexing (TDM). In FDM, dedicated channels for uplink and downlink are used, whereas in TDM, time-division is used to specify uplink and downlink slots.

3GPP Release 12 standardized device-to-device (D2D) communication between UEs, which was intended for proximity services and public safety communication during disasters. It introduced the concept of sidelink communication mode that allowed UEs to communicate directly without going through the infrastructure's base station. In this mode, a UE can perform *Discovery* where it can detect UEs in its neighborhood with or without the help of the eNodeB, and *Communication* where

the UE communicates directly with another UE without going through an eNodeB.

Release 14 [4], extended D2D communication by introducing Cellular Vehicle-to-Everything (C-V2X), which provides standardization of vehicle-to-vehicle communication through PC5 Sidelink channel. This is also referred to as LTE-V (LTE-Vehicular), and was 3GPP's response to DSRC. It allows communication between vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and vehicle-to-pedestrian (V2P) as shown in Figure 2.5.

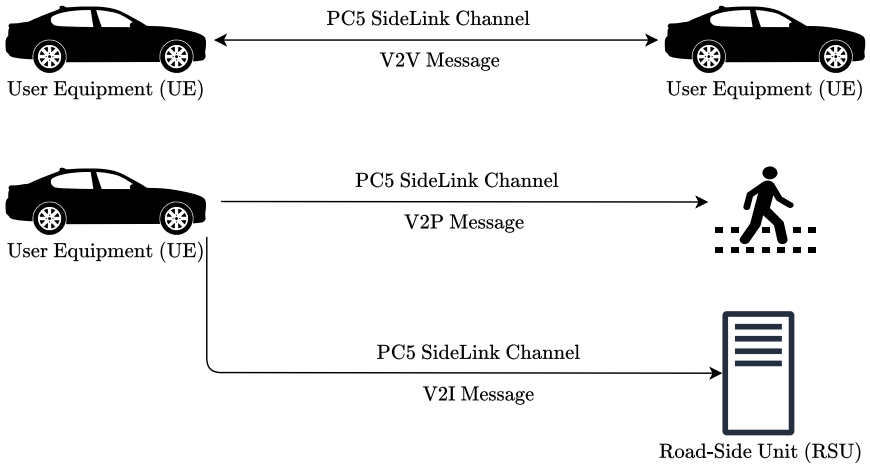


Figure 2.5: Modes of PC5 Sidelink Communication

LTE-V can theoretically operate in any frequency band, but it is commonly deployed in the 5.9 GHz band, referred to as the Intelligence Transportation System (ITS) band. It designate Physical Sidelink Shared Channel (PSSCH) for data frames and Physical Sidelink Control Channel (PSCCH) for control frames. The wireless band is divided into sub-channels referred to as transport blocks (TB). In the time domain, the channel is organized into 10ms frames, and each frame is divided into 10 1ms sub-frames. The combination of frequency and time division, illustrated in Figure 2.6, creates resources that are to be used by vehicles that want to transmit data. LTE-V has two resource allocation modes [53]:

- **Mode 3 :** UE transmitting on the sidelink receives a transmission grant from the eNB. This mode is also known as "eNB-scheduled mode". This mode is only possible when a UE is in coverage of an eNB.
- **Mode 4 :** UE selects the resource on its own, either randomly, or after sensing the medium.

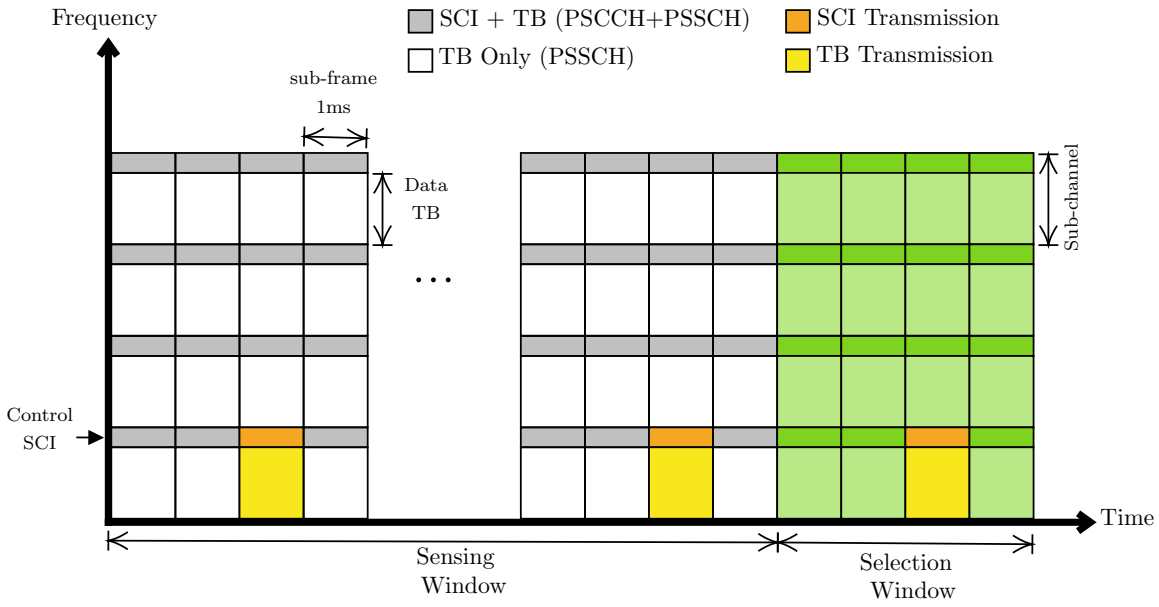


Figure 2.6: Sensing-based Autonomous Resource Selection in the C-V2X

This can be performed when vehicles are out-of-coverage of an eNB.

For the sensing-based resource selection procedure, the sensing window is 1 second, where a UE listens to see which resources are free. The sensing process is very different from the IEEE 802.11 procedure as it is done with a subframe granularity, and is based on Sidelink Control Information (SCI) and energy measurements. To reserve the resource, the transmitting UE indicates in the SCI its message periodicity for receivers to expect, so that other UEs would exclude the resource when they perform their own resource selection. This process is known as *semi-persistent scheduling* because a UE will reserve resources for a future time window before it releases the resource and repeats the process of sensing and reservation.

3GPP Release 15 [5] introduced the 5G New Radio (NR) communication in cellular networks, and Release 16 [6] extended LTE-V to use 5G NR for V2X with NR sidelink channels. It offers comprehensive and advanced vehicular communication standards with the following significant improvements over LTE-V:

- Support for unicast service with a feedback channel, Physical Sidelink Feedback Channel or (PSFCH), from a receiver UE to the transmitting UE.
- Support for hybrid ARQ (HARQ) and full support for adaptive modulation/coding (AMC).

- Seamless integration with the older LTE-V, where an NR system can schedule traffic on LTE sidelink, and LTE systems can schedule traffic on NR sidelink.

NR V2X uses similar resource allocation modes for sidelink communication with *mode 1* being guided by an eNodeB similar to mode 3 for LTE-V, and *mode 2* being autonomous resource reservation similar to mode 4 for LTE-V. In addition, NR V2X can operate in frequency range 1 (FR1) from 410 MHz to 7.125 GHz, and frequency range 2 (FR2) from 24.25 GHz to 52.6 GHz. It can also operate with different channel bandwidths, and it supports 29 modulation coding schemes indexed form `mcs-0` at the lowest end to `mcs-28` at the highest end.

For the purpose of studying 5G NR V2X alongside the 802.11-based wireless technology, we focus our work on FR1 (5.9 GHz) and channels of 10 MHz and 20 MHz bandwidth. In addition, since DSRC broadcast traffic is not acknowledged, we only consider 5G NR V2X that does not utilize a PSFCH for feedback, but rather uses a method called *blind retransmission* to increase the success of packet delivery ratio without waiting for feedback messages that introduce additional latency [7].

2.2.1 Performance Comparison with DSRC

DSRC was proposed over two decades ago in 1999, and is therefore considered older technology when compared to the more modern C-V2X which was proposed in the mid-2010s. There are extensive studies on the performance of LTE-V, 5G NR V2X compared to the IEEE 802.11-based DSRC. In [61], LTE-V is studied as a mechanism for short-range communication for vehicular networking, and concluded that LTE-V can represent an alternative to DSRC as it offers improved link budget, and can benefit from the existing cellular infrastructure for assistance. In [47], LTE V2X operating in the 5.9 GHz band is compared with DSRC to study the link-level performance. It concluded from simulations that LTE V2X can achieve the same BLER (Block Error Rate) with a lower SNR (Signal Noise Ratio) than DSRC. It also showed that LTE V2X provides a more reliable link with lower receiving power than DSRC.

In Sections 4.2, we present how 5G NR V2X can achieve improved traffic throughput with 5G NR V2X as opposed to DSRC.

2.3 Safety Applications & Periodic Broadcast

European Telecommunications Standards Institute (ETSI) oversees the process of standardization of communication protocols in Intelligent Transport Systems (ITS) in Europe. IEEE provides the physical and data-link layer specification in Europe and United States, but ETSI provides additional specifications and recommendations for application services for use case scenarios as well. IEEE 802.11 wireless frames that implements ETSI services are designated with EtherType `0x8947`, known as GeoNetworking. Packets that implement IEEE networking services exchange WSMP (WAVE Short Message Protocol) messages, distinguished by the EtherType `0x88DC`.

It is very common to refer to periodic VANET broadcast messages as BSM, Basic Safety Message. However, BSM is specified by the US-based Society of Automotive Engineers (SAE) with SAE J2735 message set, which is a closed standard. ETSI's equivalent of BSM is Cooperative Awareness Messages (CAM), which are specified by [28]. Table 2.3 lists some use case scenarios and recommendations for CAM, where messaging frequency of 10 Hz is recommended for safety applications. A higher messaging frequency may lead to congestion in the network, leading to packet loss due to collision of wireless frames attempting to use the same channel at the same time.

Use case	**Messaging Frequency (per second)**
Emergency Vehicle Warning	10
Slow Vehicle Indication	2
Intersection Collision Warning	10
Motorcycle Approaching Indication	2
Collision Risk Warning	10
Speed Limits Notification	1 to 10
Traffic Light Optimal Speed Advisory	2

Table 2.3: Uses cases of CAM

VANET applications require the use of periodic broadcast applications in vehicles to announce their presence. We used the 100ms time requirement for safety application as a baseline for evaluation. In Section 4.3, we present evaluation of the performance of periodic broadcast applications when using DSRC and 5G NR V2X.

2.4 Emergent Changes in the Technology

2.4.1 Coexistence with Wi-Fi and Unlicensed National Information Infrastructure (U-NII)

The U-NII radio band is part of the spectrum used by IEEE 802.11 radio devices as seen in Table 2.4. The U-NII-4 band was originally designated for DSRC in 1999. However, in 2020, the FCC decided to open up the lower 45 MHz of the band for Wi-Fi use after a long series of proposals for re-channelization. This would obviously impact legacy VANET applications that use the lower end of the 5.9 GHz band as many safety applications use channel 172 for "high availability, low latency" applications, which would overlap with Wi-Fi. This resulted in proposals to deal with the co-existence with Wi-Fi, where Wi-Fi would have to detect DSRC traffic and vacate the overlapping channel (Detect-and-Vacate), or use lower priority traffic (Detect-and-Mitigate) [24]. Re-Channelization (Re-Ch) of DSRC channels is another proposal, which would limit the band used for vehicular communication to 30 MHz.

Band	Frequency	Bandwidth
U-NII-1 / U-NII Indoor	5.150-5.250 GHz	100 MHz
U-NII-2A / U-NII Mid	5.250-5.350 GHz	100 MHz
U-NII-2B / U-NII Indoor	5.350-5.250 GHz	120 MHz
U-NII-2C / U-NII-2e	5.470-5.250 GHz	255 MHz
U-NII-3 / U-NII Upper	5.725-5.250 GHz	125 MHz
U-NII-4 / ITS / DSRC	5.850-5.925 GHz	75 MHz
U-NII-5	5.925–6.425 GHz	500 MHz
U-NII-6	6.425–6.525 GHz	100 MHz
U-NII-7	6.525–6.875 GHz	350 MHz
U-NII-8	6.875–7.125 GHz	250 MHz

Table 2.4: U-NII bands

2.4.2 IEEE 802.11bd and Millimeter Wave

Since the introduction of IEEE 802.11p in 1999, newer amendments to the IEEE 802.11 standard (Wi-Fi) have been proposed, namely 802.11n, 802.11ac and 802.11ax, providing support for higher orders of modulation that allows much higher throughput in a 20 MHz channel than was possible at the time of the inception of 802.11p. Table 2.5 shows data rates achievable with the newer standards. These advancements are on their way to influence vehicular communication when they are introduced by the upcoming 802.11bd amendment, which will provide enhancements for Next-Generation V2X

(NGV), and inherits the physical layer capabilities of 802.11ax. The newer enhancement aim to provide increased throughput and maintain interoperability with existing 802.11p devices. To ensure compatibility, 802.11bd must be able to decode at least one mode of 802.11p's communication modes and vice versa.

The new IEEE standard 802.11ay specifies the use of Wi-Fi operations in the 60 GHz band [40], over a 14 GHz band (from 57 GHz to 71 GHz), also known as millimeter wave frequencies. It is proposed to be divided into 6 non-overlapping channels, each occupying 2160 MHz of spectrum, and providing 1760 MHz of bandwidth as shown in Table 2.6. Such high bandwidth can achieve higher bit rates, and devices supporting 60 GHz are referred to as Directional Multi-Gigabit (DMG) stations. The 802.11bd amendment aims to support operation in the 60 GHz band, with proposals for possible support of OCB mode in the 60 GHz band as well, with the intention of using it to support very high throughput use cases in that band, and reserving the 5.9 GHz band for reliable communication, typically used by safety applications.

2.5 CACC & Packet Loss Behavior

Longitudinal control of a platoon of vehicles has been extensively reported upon in the literature in the past decades. The objective of CACC is to set a spacing policy between vehicles of a CACC platoon using on-board sensors and a wireless communication network. The on-board sensors can measure the distance and relative velocity from another vehicle. Wireless communication is used to communicate information between vehicles, and it can follow various structures. Early designs

Modulation	**Channel Width (Hz)**	**Highest data rate (Mbps)**	**Notes**
High Throughput (HT)	20	288.8	802.11n and newer
	40	600	
Very High Throughput (VHT)	20	346.8	802.11ac and newer
	40	800	
	80	1733.2	
	160	3466.8	
High Efficiency (HE)	20	1147	802.11ax and newer
	40	2294	
	80	4804	
	160	9608	

Table 2.5: Newer Modulations for newer IEEE 802.11 standards

Channel	Center Frequency (GHz)	Min.	Max.
1	58.32	57.24	59.40
2	60.48	59.40	61.56
3	63.64	61.56	63.72
4	64.80	63.72	65.88
5	66.96	65.88	68.04
6	69.12	68.04	70.20

Table 2.6: IEEE 802.11ay in the 60 GHz band.

of CACC models such as [67] suggest a bidirectional communication topology, where each vehicle makes use of information receives from its direct preceding and following vehicles. More recent work, such as California's PATH program [92], suggests that the use of a bidirectional information flow structure is unnecessarily complicated, and simplified the problem with a unidirectional approach in which a vehicle within a platoon is only concerned with information received from preceding vehicles. With unidirectional information flow, several control models were proposed, such as multi-vehicle look-ahead controllers[81], where information received from multiple preceding vehicles are used in addition to information from the platoon leader. In [50], it was shown that a higher traffic throughput achieved by a reduced time headway is possible by only using information from two preceding vehicles, without the need of information from the platoon leader. In [69], a one-vehicle look-ahead CACC controller was proposed, showing that it is possible to reduce the time headway gap well below 1 seconds while maintaining string stability.

The technology used for vehicular communication, DSRC, does not offer performance guarantee, and therefore, attention was given to studying CACC under different network conditions. CACC makes use of acceleration information received from the preceding vehicle over the communication network. The work in [70] proposes that in case of network impairment, a CACC application shall fall back to a degraded CACC state, dCACC, in which vehicles estimate the acceleration of preceding vehicles from on-board sensors. While their work offers improvement over a traditional ACC system, it does not achieve string stability that a CACC controller with no loss can achieve.

It is important to understand the nature of packet loss events exhibited by a VANET. This allows us to create realistic packet loss scenarios that CACC applications may encounter. In [18], the authors characterized packet loss events in 802.11p/WAVE vehicular networks that uses multi-channel operations to derive a function to predict loss. They found that due to channel switching, packets might be dropped following a repeated pattern. In [33], the 802.11p communication channel is modeled

based on Discrete Time Markov Chain (DTMC), where packet loss status is modeled with state machines of good states, where packet reception is uninterrupted, and loss states, where packet loss is expected to occur with high probability. In [16], packet loss in VANETs is assumed to be bursty, and use a two-state Markov model to predict probability of packet loss based on Signal-to-noise ratio.

There is a consensus in the literature that packet loss occurrences are not independent. That is, they do not follow a Bernoulli process in which packets would be dropped with a certain probability, but should rather be expressed in a way similar to the Gilbert-Elliott model. In VANETs, packet loss events exhibit temporal and spacial correlation. Therefore, we focused our testing on losses of bursty nature rather than assigning an independent probability of loss to each packet.

In order to study network impairment and fallback method performance, we implemented a simulation based one a one-vehicle look-ahead CACC controller. Since distance and relative velocity can be obtained using on-board sensors in case of network impairment, acceleration of the preceding vehicle can be estimated using a Kalman filter. In our study, we simulated several packet loss scenarios and their effect on string stability. We propose adapting the time headway value parameter of the CACC controller according to a network reliability metric.

VANETs can utilize different radio access technologies such as 802.11-based DSRC and Cellular Vehicle-to-Everything (C-V2X), and amendments are proposed to enhance the standards for use in VANETs, such as the P802.11bd that adds support to the 60 GHz band. Our proposed solution can be applied to any radio access technology that can achieve the required low latency for CACC.

2.6 Tracking Moving Objects Using a Kalman Filter

Vehicles are equipped with on-board LiDAR sensor to detect vehicles and objects on the road. A sensor reading involves a small amount of noise which is inherent to sensors' design. If we modeled sensor noise as a zero-mean Gaussian function with a variance $\sigma^2 = 0.1$ (Generally, $\mathcal{N}(\mu, \sigma^2)$), then the sensor reading for distance alone is acceptable as it is within a variance of 0.1 meters (10 cm). This error in measurement is referred to as *measurement noise*. The distance sensor reading is needed for developing a mitigation strategy, where CACC falls back to sensor-estimated acceleration when the network condition is impaired. If vehicle i is following vehicle $i-1$, then the observed

relative velocity $\vec{v}_{i|i-1}$ can be obtained by the change in distance between two consecutive distance sensor readings over a sensor read interval time Δ_m

$$\vec{v}_{i|i-1}(k) = \frac{d_{i-1}(k) - d_{i-1}(k-1)}{\Delta_m} \tag{2.1}$$

Similarly, acceleration can be measured by computing the relative acceleration $\vec{a}_{i|i-1}$ can be obtained using three consecutive distance reading, or simply by using the last two relative velocity values

$$\vec{a}_{i|i-1}(k) = \frac{\vec{v}_{i|i-1}(k) - \vec{v}_{i|i-1}(k-1)}{\Delta_m} \tag{2.2}$$

$$= \frac{d_{i-1}(k) - d_{i-1}(k-1)}{\Delta_m} - \frac{d_{i-1}(k-1) - d_{i-1}(k-2)}{\Delta_m} \tag{2.3}$$

We can then find the acceleration of vehicle $i-1$ by adding vehicle i's acceleration to the relative acceleration observed of vehicle $i-1$

$$\ddot{x}_{i-1}(k) = \vec{a}_{i|i-1}(k) + \ddot{x}_i(k) \tag{2.4}$$

A moving object changes its position, velocity and acceleration constantly. Therefore, the estimations using Equation 2.1 and 2.4 involve a small amount of error. This is known as *Process Noise*, as it would be present even if sensor readings were 100% perfect due to the object's changing velocity and acceleration between sensor readings.

The aforementioned *measurement noise* produces distance values with a variance of 0.1 meters. However, when we use those sensor readings to estimate relative velocity and acceleration, the error amplifies, producing unacceptable results as shown in Figure 2.7. Therefore, we need a better way to estimate a preceding vehicle's acceleration that simply uses three consecutive readings, and Kalman filters can offer significant improvement to the accuracy of acceleration estimation of a moving object.

Kalman filters [30] can be used to estimate the state of an object while taking into consideration the presence of measurement noise and process noise. It has been used extensively in the literature to track moving objects such as tracking for automated surveillance systems [66], robotic automation [19] and aircraft tracking [9]. In the area of vehicular networks, a Kalman filter is applied to predict the location of a vehicle accurately using real vehicles' mobility traces and model-driven traces [34]. In [39], an accurate algorithm was developed for vehicle positioning using a Kalman filter. For

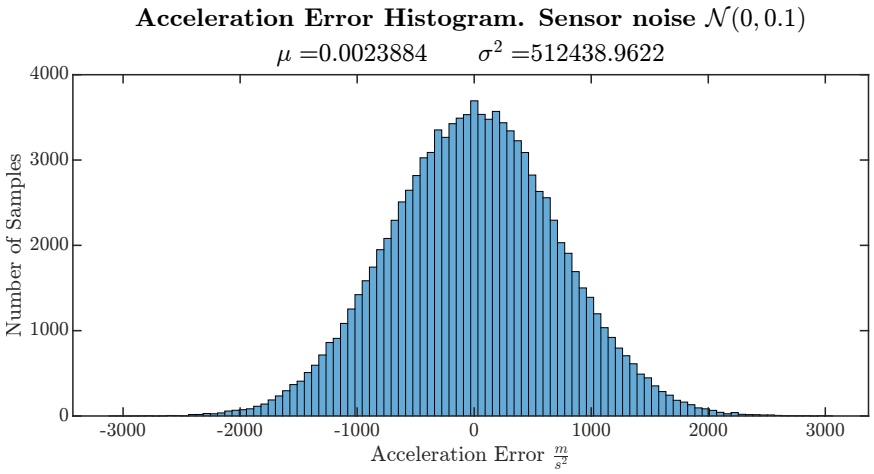


Figure 2.7: Unacceptable acceleration estimation due to measurement & process noise

CACC platoons using wireless communication to exchange acceleration information, Kalman filters are applied to estimate a preceding vehicle's acceleration when communication is lost [70][89]. In this work, we study CACC platoons under varying network impairment scenarios, and we use a Kalman filter to estimate acceleration when communication is lost. In Section 5.1, we propose an improved Kalman filter that improves acceleration estimation when compared to Rayamajhi's implementation in [74], and only requires input from a single distance sensor.

Chapter 3

CACC Control Modeling

A CACC controller requires the presence of a wireless communication network over which vehicles can broadcast information periodically. This information can include acceleration, velocity and position. However, if vehicles rely solely on information received over a wireless network, there is no fallback strategy if the wireless network becomes unavailable. Therefore, vehicles can use local on-board sensors such as LiDAR to detect the distance and relative velocity of the preceding vehicle.

3.1 CACC Controller Model

We implemented an ns-3 simulation of a one-vehicle look-ahead CACC model as shown in Figure 3.1. The CACC model used is based on [69]. If we have a CACC platoon of m vehicles, a spacing policy for vehicle i can be defined with

$$d_{r,i}(t) = r_i + hv_i(t), \qquad 2 \leq i \leq m \tag{3.1}$$

where h is the desired time headway. r_i is the stand-still reference distance. The spacing error $e_i(t)$

$$\begin{aligned} e_i(t) &= d_i(t) - d_{r,i}(t) \\ &= (s_{i-1}(t) - s_i(t) - L_i) - (r_i + hv_i(t)) \end{aligned} \tag{3.2}$$

where $s_i(t)$ is the position of vehicle i at time t, and L_i is the length of vehicle i.

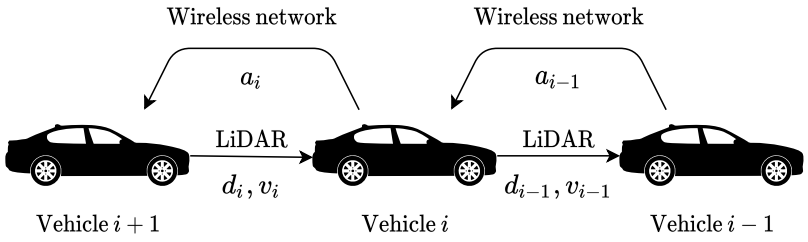


Figure 3.1: One-Vehicle Look Ahead CACC

The CACC controller for a following vehicle i updates its acceleration every t_{CACC} period of 100ms based on the following

$$\dot{u}_i(t) = -\frac{1}{h}a_i + \frac{1}{h}\left(k_p e_{1,i} + k_d e_{2,i}\right) + \frac{1}{h}u_{i-1} \tag{3.3}$$

where a_i is the current vehicle's acceleration, $e_{2,i}$ is the change in $e_{1,i}$, namely $\dot{e}_{1,i}$, and k_p, k_d are gain parameters set to 0.66 and 0.7 respectively. The value of u_{i-1} is the perceived acceleration of the preceding vehicle, which is either received over wireless communication or estimated from on-board sensor reading when the network is impaired. The CACC controller also limits the acceleration values to $\pm 12\frac{\text{ft}}{\text{s}^2}$, which is equal to $\pm 3.6576\frac{\text{m}}{\text{s}^2}$ to avoid harsh accelerations and deceleration for passengers' comfort.

The value $\dot{u}$ is the change in acceleration from a starting acceleration a_{start} to a target acceleration a_{target}, which is performed linearly over a CACC update interval t_{CACC}. For example, at time t_{start}, vehicle i computes a new target acceleration a_{target}, and changes its current acceleration linearly until it reaches the target acceleration at $t_{\text{target}} = t_{\text{start}} + t_{\text{CACC}}$, where a new target acceleration would be computed again, and so on. The value u_{i-1} that vehicle $i-1$ broadcasts over the wireless network can either be its current acceleration, or its target acceleration computed from Equation 3.3

Vehicles' absolute positions and velocities are updated accordingly as well, following a piecewise acceleration whose slope j_i is computed, every time a new acceleration target is computed, as:

$$j_i = \frac{a_{\text{target}} - a_{\text{start}}}{t_{\text{CACC}}} \tag{3.4}$$

and then at a time $t \in [t_{\text{start}}, t_{\text{target}}]$, vehicle i's acceleration $a_i(t)$, velocity $v_i(t)$ and position $d_i(t)$

are updated continuously according to the following motion equations:

$$a_i(t) = a_i(t_{\text{start}}) + j_i \Delta t \tag{3.5}$$

$$v_i(t) = v_i(t_{\text{start}}) + a_i(t)\Delta t + j_i \frac{\Delta t^2}{2} \tag{3.6}$$

$$d_i(t) = d_i(t_{\text{start}}) + v_i(t)\Delta t + a_i(t)\frac{\Delta t^2}{2} + j_i \frac{\Delta t^3}{6} \tag{3.7}$$

where Δt is $t - t_{\text{start}}$. The platoon leader follows an acceleration profile that is read from a text file, where the leader starts from a velocity and acceleration of 0, and obtains a new target acceleration from the file every 100ms, following the aforementioned motion equations.

3.1.1 String Stability

In order to evaluate the performance of a CACC deployment, we need to determine whether the platoon is *string stable*. For a string stable platoon, the amplification of disturbances in upstream direction of either distance error, velocity or acceleration is diminished before they reach the end of the platoon. In this section we review string stability from the perspective of [69, 71]. In Section II of both [69] and [71] the authors present a brief discussion of the concepts of string stability. They identify three general approaches that have been applied. The first is a formal approach based upon Lyapunov stability. This approach is said to suffer from constraints that make it lack practicality in real-world systems. Specifically, external disturbances, including speed variation of the lead vehicle, are said to be not supported. In a second approach, the Z-transform is used in analysis of infinite strings. The authors point out in [71] that "Unfortunately, the stability properties of finite-length strings, being practically relevant, might not converge to those of infinite-length strings as length increases. This can be understood intuitively by recognizing that in a finite-length platoon, there will always be a first and a last vehicle, whose dynamics may significantly differ from those of the other vehicles in the platoon, depending on the controller topology. Consequently, the infinite-length platoon model does not always serve as a useful paradigm for a finite-length platoon as it becomes increasingly long."

The third approach is referred to as a *performance-oriented* approach. A finite string of N vehicles

is *defined* to be string stable if and only if

$$||z_i(t)||_{\mathcal{L}_2} \leq ||z_{i-1}(t)||_{\mathcal{L}_2} \ \ \forall t \geq 0 \text{ and } 2 \leq i \leq N \tag{3.8}$$

where $z_i(t)$ can be (with equivalent results) the positional error $e_i(t)$ of vehicle i with respect to vehicle $i-1$, its speed $v_i(t)$, or its acceleration $a_i(t)$.

This metric also includes an assumption that the initial state of the string is stationary. Consequently, it is not applicable to strings whose membership may dynamically change nor to analysis of sub-intervals of string behavior.

Nevertheless, for simulation studies, it is clear that this metric is readily applied to any finite length platoon with any combination of controller inputs and the required initial conditions present no obstacle. Therefore, in this work we also employ this specific characterization of string stability and use $v_i(t)$ in our computations.

This stability condition is a retrospective analysis of the platoon's stability, and can be expressed as a single value that must satisfy the following condition for a stable platoon:

$$\frac{||v_i(t)||_{\mathcal{L}_2}}{||v_{i-1}(t)||_{\mathcal{L}_2}} \leq 1 \tag{3.9}$$

3.2 Fallback mitigation

The CACC controller uses periodic acceleration updates received over the wireless network. When the network becomes unavailable, these updates are lost, and therefore the model does not receive updated acceleration inputs. The CACC controller in each vehicle computes a new target acceleration every 100ms, and it also expects to receive acceleration updates from the preceding vehicle every 100ms. If acceleration updates were not received, the controller will use the last known acceleration value until it detects packet loss. In VANET based on DSRC/WAVE, packet loss is not explicitly detected, and it is possible to occasionally miss some acceleration updates due to frame collisions. The controller will use the last known acceleration from preceding vehicle as long as the last update received was less than 300ms ago. If a vehicle does not receive an acceleration update after 300ms, it assumes that the communication network is in a loss state, and switches to a fallback mode, where

acceleration of the preceding vehicle is estimated using on-board sensors.

The fallback mode works in the absence of the wireless network by using on-board sensors and a Kalman estimation filter to estimate the preceding vehicle acceleration as detailed in the work in [74]. In ns-3, we can obtain the exact distance and relative velocity between two simulation nodes. However, this is unrealistic as real-world sensors have inaccuracies within a certain margin of error. Therefore, we apply a zero-mean Gaussian noise to the actual distance between simulation nodes, such that distance error is within ± 0.3 meters, and relative velocity accuracy is within +2 km/h and -3 km/h. These margins of errors are based on the accuracy requirement for approval by National Highway Traffic Safety Administration (NHTSA) [64]. We assume that the variance in distance error σ_d^2 and the variance in relative velocity error σ_v^2 to be 0.1 in our simulation. We assume that distance and velocity readings are obtained independently, and refer to the added distance and relative velocity sensor noises as $\mathcal{N}_{dist}$ and $\mathcal{N}_{vel}$ respectively. A Kalman filter has two sub-processes executing recursively in a loop: *prediction phase* which periodically predicts the state variables, and correction phase which attempts to correct the predicted state variable values using the measurement. If x is the state of a system to be estimated, u is control input, and z is the measurements done periodically, then a Kalman filter is designed to address the general problem of estimating the state $x_k \in \mathcal{R}^n$ with measurements $z \in \mathcal{R}^m$ of a process defines by the discrete time equations:

$$x_k = \mathbf{A}x_{k-1} + \mathbf{B}u_{k-1} + w_{k-1} \tag{3.10}$$

$$z_k = \mathbf{H}x_k + v_k \tag{3.11}$$

The random variables w_k and v_k are assumed to be independent zero-mean Gaussian random variables that represent process and measurement noise respectively. The $n \times n$ matrix $\mathbf{A}$ relates the past estimate of the state x_{k-1} to the current x_k in the absence of noise and control input u_k. Similarly, the $n \times 1$ matrix $\mathbf{B}$ relates the control input u_k to the state x_k. The $m \times n$ matrix $\mathbf{H}$ relates the state x_k to the measurement variables z_k. In practice, the matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{H}$ vary with time, but we assume they remain constant for our study.

We also initialize $\mathbf{P}$ as a 4x4 matrix with all elements set to 1. The Kalman filter's prediction phase:

$$\hat{x}_k^- = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}\hat{u}_{k-1} \tag{3.12}$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \tag{3.13}$$

where $\hat{x}_k^-$ is the priori estimate of the state x_k at time step k and $\hat{x}_k$ is the posteriori estimate of state x_k at time step k. $\mathbf{P}_k^-$ is the priori estimate error covariance and $\mathbf{P}_{k-1}$ is the posteriori estimate error covariance. The goal is to find an equation that computes a posteriori estimate $\hat{x}_k$ as a linear combination of an a priori estimate $\hat{x}_k^-$ and a weighted difference between an actual measurement z_k and a prediction $\mathbf{H}\hat{x}_k^-$. The Kalman *correction phase* is defined by:

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_k^-\mathbf{H}^t + \mathbf{R})^{-1} \tag{3.14}$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \tag{3.15}$$

$$\hat{x}_k = \hat{x}_k^- + \mathbf{K}_k(z_k - \mathbf{H}\hat{x}_k^-) \tag{3.16}$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k^- \tag{3.17}$$

where the $n \times n$ matrix $\mathbf{K}_k$ is the gain that minimizes a priori error covariance $\mathbf{P}_k^-$ at time step k.

The relative distance from vehicle $i-1$ to preceding vehicle i at time k is computed with the addition of sensor noise as follows:

$$d_i[k] = x_{i-1}[k] - x_i[k] + \mathcal{N}_{dist} \tag{3.18}$$

Similarly, the relative velocity between vehicles $i-1$ and i at time k is computed by adding the noise value for speed measurement as follows:

$$r_i[k] = v_{i-1}[k] - v_i[k] + \mathcal{N}_{vel} \tag{3.19}$$

Between two consecutive sensor readings, we will define distance traveled by vehicle i as $s_i[k]$, and using two consecutive sensor readings, we can estimate the distance traveled by preceding vehicle $i-1$ as

$$\hat{s}_{i-1}[k] = s_i[k] + d_i[k] - \hat{s}_{i-1}[k-1] \tag{3.20}$$

where $\hat{s}_{i-1}[k-1]$ is the prior estimate of distance traveled by the leading vehicle i-1. The velocity

of preceding vehicle $i-1$ is estimated from the relative velocity reading and the velocity of vehicle i as follows

$$\hat{v}_{i-1}[k] = r_i[k] + v_i[k] \tag{3.21}$$

The acceleration of the preceding vehicle $i-1$ is estimated using the change in estimated velocity between two sensor readings. Let t_{dme} be the time between two sensor readings, acceleration is estimated as:

$$\hat{a}_{i-1}[k] = \frac{\hat{v}_{i-1}[k] - \hat{v}_{i-1}[k-1]}{t_{\text{dme}}} \tag{3.22}$$

We will use the term $\hat{m}_{i-1}[k]$ to denote the approximate rate of change of preceding vehicle's acceleration at time k, and therefore, we establish the state variable $x[k]$ as:

$$x[k] = \begin{bmatrix} m_{i-1}[k] \\ s_{i-1}[k] \\ v_{i-1}[k] \\ a_{i-1}[k] \end{bmatrix} \begin{matrix} \text{Denotes change in acceleration, or } \textit{jerk} \\ \text{Denotes } \textit{distance traveled} \\ \text{Denotes } \textit{velocity} \\ \text{Denotes } \textit{acceleration} \end{matrix} \tag{3.23}$$

The Kalman filter model parameters at any time step k are given by defining the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{H}$ and u as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \dfrac{1}{t_{\text{dme}}} \\ \dfrac{t_{\text{dme}}^3}{6} & 0 & t_{\text{dme}} & \dfrac{t_{\text{dme}}^2}{2.0} \\ \dfrac{t_{\text{dme}}^2}{2.0} & 0 & 1 & t_{\text{dme}} \\ t_{\text{dme}} & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

We also defined the measured estimated state of preceding vehicle $i-1$, denoted with z as follows:

$$z = \begin{bmatrix} \hat{m}_{i-1}[k] \\ \hat{s}_{i-1}[k] \\ \hat{v}_{i-1}[k] \\ \hat{a}_{i-1}[k] \end{bmatrix}$$

We use two covariance matrices $\mathbf{Q}$ and $\mathbf{R}$, which in practice would be provided by a sensor's manufacturers, but since our study is simulation-based, we derive $\mathbf{Q}$ and $\mathbf{R}$ to be in line with the error variance of 0.1 as follows:

$$\mathbf{Q} = \begin{bmatrix} 0.1^{-4} & 0 & 0 & 0 \\ 0 & 0.1^{-5} & 0 & 0 \\ 0 & 0 & 0.1^{-5} & 0 \\ 0 & 0 & 0 & 0.1^{-5} \end{bmatrix},$$

$$\mathbf{R} = \begin{bmatrix} 0.2 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix}$$

The estimated acceleration value of the preceding vehicle is extracted from the $\hat{x}_k$ vector, and is always calculated with every sensor reading. If a vehicle missed three consecutive acceleration updates through the wireless network, it will use the most recent acceleration estimated by the Kalman filter described here. Without using a Kalman filter, it is still possible to estimate the acceleration of the preceding vehicle, however our tests show that using a Kalman filter produces smaller error in estimation as shown in Figure 3.2.

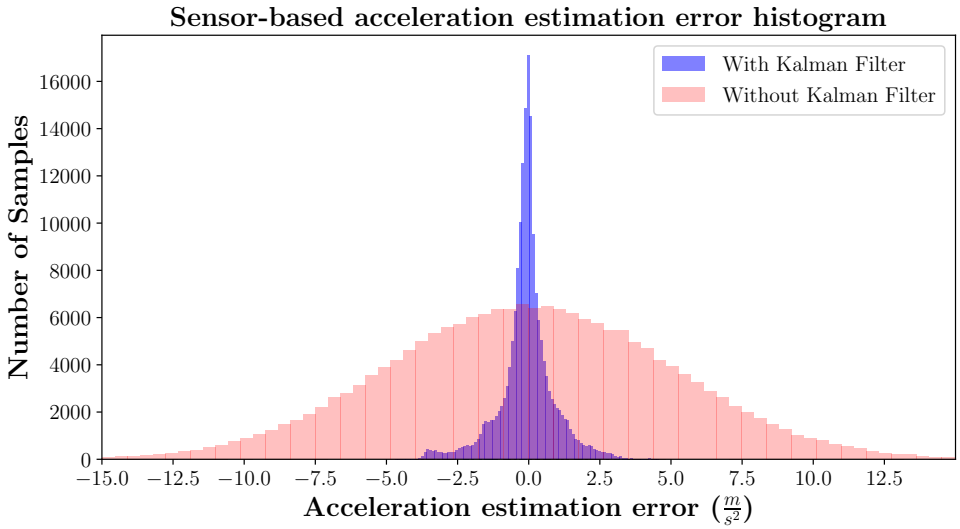


Figure 3.2: An example of improvement in acceleration estimation when using a Kalman filter

3.3 Simulation Setup

We use ns-3 network simulator to create a one-vehicle look-ahead CACC application. We created 10 vehicles, a platoon leader and nine followers. The simulation starts with stationary nodes separated by a given stand-still reference distance. The platoon leader is controlled directly by feeding it new acceleration values every 100ms from a preset acceleration profile. This simulates a real vehicle driven by a human driver, or an autonomous self-driving vehicle. The remaining vehicles in the platoon calculate their acceleration based on the perceived acceleration of the preceding vehicle, and the distance measured by on-board sensors. Every vehicle has two independent simulation loops that are repeated every 100ms: the first loop broadcasts the vehicle's acceleration, and the second loop computes new target accelerations using Equation 3.3, where it uses the last known perceived acceleration of the preceding vehicle.

We can get exact distance and relative velocity between two simulation nodes in ns3, but we add noise to this data to simulate error in sensor readings. To study the effect of burst loss on platoon safety, we used synthetic acceleration profiles that follow linear, sinusoidal, and step function. We also used two profiles obtained from real vehicles, namely the *us06* drive cycle which was obtained from the US Environmental Protection Agency (EPA), and *c2a*, a profile that was generated by driving a car from Clemson, SC to Anderson, SC. To study our proposed solution, we needed longer

Beacon Interval	100 ms
CACC Interval	100 ms
Number of vehicles	10 vehicles
Velocity range	0 to 35 m/s
Acceleration range	-3.6576 to 3.6576 m/s^2
K_p	0.66
K_d	0.7
Sensor interval	25ms

Table 3.1: ns-3 simulation parameters

drive cycle data, so we used five acceleration profiles obtained from real vehicles that we acquired from the National Renewable Energy Laboratory (NREL) [1].

A simulation stops if a vehicle collision occurred or when the specified total simulation time is reached. After a simulation concludes, the observed string stability and traffic flow is generated. To compute traffic flow for a platoon of N vehicles, let $x_1(t)$ and $x_N(t)$ be the positions of the platoon leader and the N-th vehicle at time t respectively. Let t_e be the time at which the N-th vehicle passes the point $x_1(t)$. Traffic flow $\mathcal{F}(t_e)$ at time t_e can be defined as:

$$\mathcal{F}(t_e) = \frac{N}{t_e - t} \quad \text{vehicles per second} \tag{3.24}$$

We repeat traffic flow computation every time the N-th vehicle passes a position where the platoon leader once was until the end of a simulation run. Simulation results are acceptable only if they resulted in a string stable platoon, and did not cause a collision. It is common to express traffic flow in *vehicles per hour*, and therefore we will multiply the result by 3600.

3.4 Network Reliability Assessment & Time Headway Value

We studied how CACC platoons react to different packet loss processes. We noticed that when modeling loss events as a Bernoulli process, the CACC is resilient even with high packet loss ratios. However, since previous work provided solid evidence that packet losses are bursty, we needed to test how would CACC performs under these conditions. Not surprisingly, CACC performance is influenced by the length of bursts of packet loss. Smaller burst losses of 10 packets (lasting for 1 second) do not impact the stability of a CACC platoon, allowing us to set a lower time headway value h. As we tested with longer burst losses, the platoons became unstable, requiring the use of

higher values of h to maintain string stability. We call the value of h that ensures stable operation the *minimum string stable headway*. When burst length goes above 300 packets (30 seconds), the minimum string stable headway value remains more or less constant as the increase in headway is sufficient for the fallback method to work without causing instability as shown in Figure 3.3.

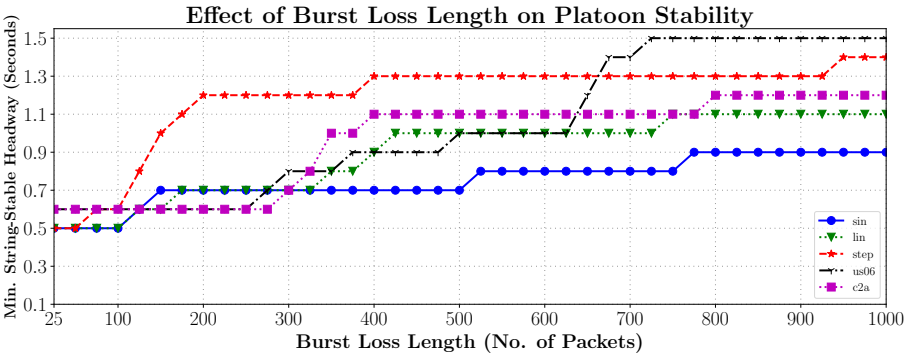


Figure 3.3: Packet burst length effect on platoon stability for different synthetic acceleration profiles

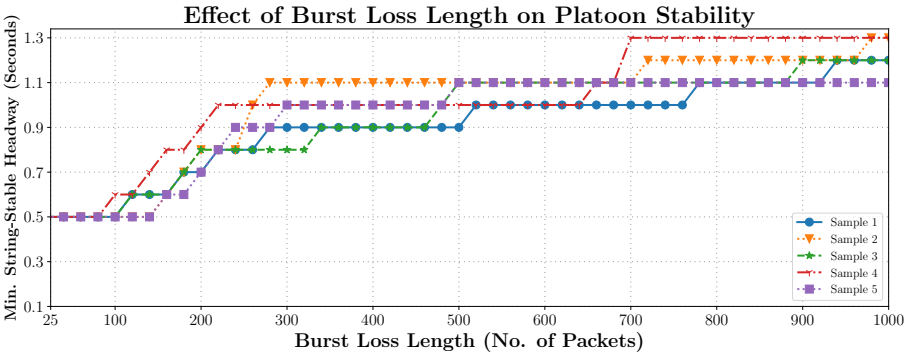


Figure 3.4: Packet burst length effect on platoon stability for different real vehicle acceleration profiles obtained from NREL

We seek to assess the network reliability to base our decision of adapting the h value in order to achieve string stable platoon operation and maximize traffic flow. Packet delivery ratio (PDR) would be feasible if the loss process followed a Bernoulli process where every single loss event is independent. However, we needed to devise a mechanism based on the bursts of packet losses. We devised a network reliability metric, which corresponds to the effect of long loss lengths on the minimum string stable headway value as follows

$$r(B) = 1 - \tanh\left(\frac{B \times M}{175}\right) \tag{3.25}$$

where B is the length of the maximum sustained burst of packet loss that the network experienced, tanh is the hyperbolic tangent function, and M is a scaling multiplier that our algorithm can control to make the criteria more or less conservative in estimating reliability. The default value of M is 1, but can change during simulation. The choice of the tanh function in this manner provides a graph that is inversely proportional to the effect burst of packet losses, and therefore a good fit to assess reliability as it pertains to CACC platoon stability.

Vehicles of the CACC platoon start in a *no loss* state, where all packets are received as expected every 100ms. If a loss is detected, a vehicle will enter into a *burst loss* state in which the maximum burst length B is tracked and used as an input to the function $f(B)$, defined in Equation 3.26, to adapt the value of the time headway value h. When packet reception resumes, we could have multiple scenarios about the reception after loss event:

1. The reception is very short, meaning that network reliability has not improved significantly. As a result, subsequent losses are added to the maximum burst loss B.
2. The reception is moderate, so we will not add subsequent losses to the maximum burst loss B, and we will start a new counter, maintaining the value of B
3. The reception is long enough to assume that the network condition has improved, so we move to a *recovery state*.

This is illustrated in the state-machine in Figure 3.5, where p and q are used as constant parameters to define the thresholds that dictate the transitions. In our simulations, we set $p = 2$ and $q = 5$. This means that we have to receive consecutive packets in a burst that is five times the size of the largest burst before we assume that the network is back to perfect condition. In addition, if good burst between losses is two times the loss length, we will set the multiplier M to 0.8, since we observed that this kind of loss is more forgiving on CACC string stability. If good packet burst between loss bursts is shorter than 25% of B, we set the multiplier M to 4, and if it is less than 10%, we set the M to 20 as we have observed that short good bursts of packets between significantly longer burst losses have severe impact on the string stability of CACC operation. We use the value of B, which we define as the maximum burst loss length, to compute a new higher headway value to mitigate

packet loss events as follows:

$$h = f(B) = (L - U)r(B) + U \tag{3.26}$$

Where L and U are the lower and upper bounds on the value of the time headway value h. We notice, as shown in Figure 3.3, that for all acceleration profile, an upper bound U of 1.5 seconds is sufficient with very long packet burst loss, that is, it can be used when there is a total network outage while maintaining string stability. The experiments also show that $L = 0.5$ can be used as a lower bound for time headway for the NREL acceleration profiles when the network is perfect, as shown in Figure 3.4. The value of B is limited to a maximum of 300, where we assume that network reliability is zero when $B = 300$.

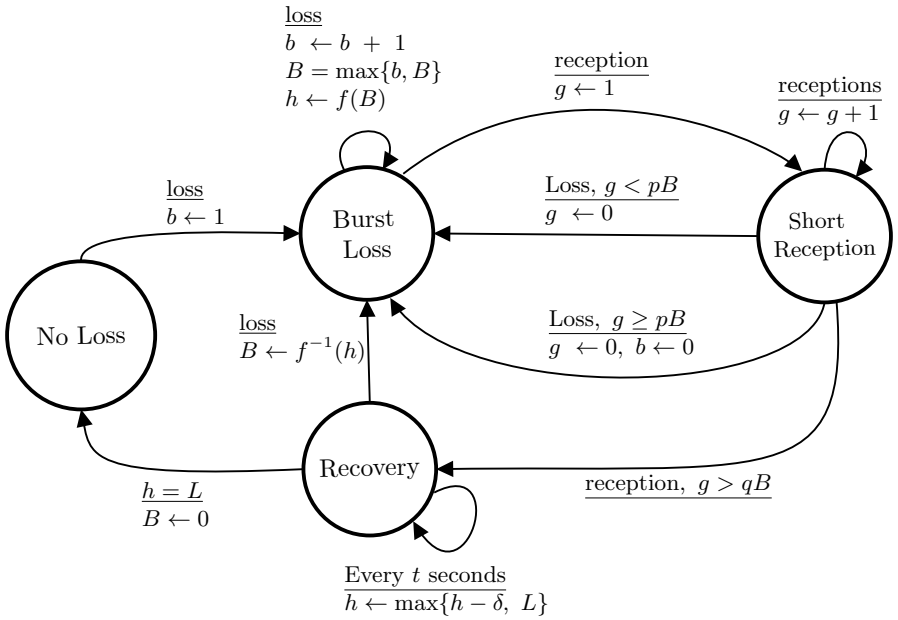


Figure 3.5: State-Machine for evaluating network reliability for CACC.

Once we reached *recovery state*, the value of the time headway h is reduced. Our initial experiments determined that reducing h too quickly could result in instability or a collision. Therefore, the process is performed gradually by δ over an update period t. We choose a value of $\delta = 0.05$. In addition, we also reduce h in a front-to-back order; that is, vehicles in the front of the platoon reduce their time headway earlier than the vehicles in the back, and over shorter interval of time, which we found helps maintain platoon's string stability.

3.5 Packet Loss Scenarios

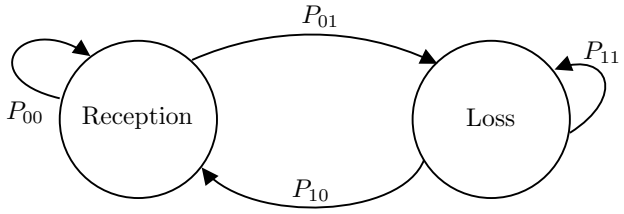


Figure 3.6: Modeling Packet Loss Scenario

Packet loss is modeled using a state-machine was two states, a *reception state* where all packets are received, and a *loss state* where packets are lost as shown in Figure 3.6. The switching between a reception and loss state is stochastically controlled. If we use 0 and 1 denote the reception and loss states respectively, we can define P_{ij} as the probability to transition to state j when at state i. Those probabilities can be represented with a 2x2 matrix. We can parameterize the loss process with two parameters: the Mean Good Length (MGL), which represents the mean number of received packets in a reception state, and the Mean Burst Length (MBL), which represents the mean number of lost packets in a loss state, and we compute the values of $P_{00}, P_{01}, P_{11}, P_{10}$ as follows:

$$P_{00} = \frac{MGL - 1}{MGL}, \quad P_{01} = \frac{1}{MGL}, \quad P_{11} = \frac{MBL - 1}{MBL}, \quad P_{10} = \frac{1}{MBL}$$

We set up our CACC simulation to start with an ideal wireless network connectivity with no packet loss, and later we trigger a loss event to study the impact of different loss processes on the performance of CACC. We tested using three loss processes shown in Table 3.2. A denial-of-service (DoS) attack can take an on-off pattern [57], so we modeled it as a loss process with $MGL = 50, MBL = 50$. The network outage scenario represents a condition where communication is lost completely due to temporary malfunction in equipment, or a complete denial-of-service (DoS) attack. To model loss in a congested network, we performed simulation test with many vehicles using the same communication channel and broadcasting every 100ms, and observed that loss takes patterns of longer bursts followed by shorter reception bursts.

In a simulation where there is no packet loss, the value of the time headway h can be set very low, allowing for higher vehicular traffic throughput. If we chose a very small h value, and then triggered packet loss, it is possible for instability or a vehicle collision to occur.

Therefore, when packet loss occurs, we use the reliability metric discussed earlier to renew the time headway with a higher value, which effectively increases the separation distance between vehicles, and consequently, reduces the risk of instability or collisions.

Loss Process	**MGL**	**MBL**
Congestion	10	40
On/Off Denial of Service	50	50
Network outage	1	10000

Table 3.2: Packet Loss Processes Used in Testing

The loss process can be applied to vehicles individually, and due to its stochastic nature, different vehicles would exhibit different loss lengths which would result in different reliability metric observed by different nodes, and hence different headway values chosen by the vehicles. To mitigate that, we make a simplifying assumption that every vehicle would broadcast its current observed network reliability metric value along with the acceleration. Moreover, it is possible to assume that since the communication range of conventional DSRC devices reaches 1000 meters, all vehicles are able to receive frames intended for other vehicles, and therefore, can compute the lowest reliability observed by a member of the platoon. Alternatively, CACC vehicles can also use an alternative WAVE channel to broadcast reliability or use cellular LTE connection to share the network reliability value. This allows every platoon member to have a global view of reliability. It is possible to consider only local reliability metrics, but that requires a more conservative network reliability to achieve string stability under loss.

3.6 Dynamic Adjustment of Headway Value

The idea of our work is to allow our CACC platoon to use the minimum string stable headway value when the network condition is perfect. Our tests with ten vehicles with platoon leader following a drive cycle obtained from NREL shows that the minimum string stable headway to use is $h = 0.5$. When the network condition is degraded, vehicles resort to estimating the acceleration of their respective preceding vehicle using on-board sensors which is not as accurate as receiving acceleration updates over the wireless network. This has the potential of increasing the risk of instability or collision when a low time headway value is used. Our tests with ten vehicles showed that for $h \geq 1.5$, we can achieve stable platoon operation with sensor-estimated acceleration alone, i.e.

without a wireless network.

Our headway adaptation algorithm sets the headway value such that $h \in [0.5, 1.5]$. In a trivial loss scenario where we run with a perfect network condition, followed by a period of total network outage before resuming perfect operation, one would assume that adjusting the value of h from 0.5 to 1.5 and then back to 0.5 would suffice as a countermeasure. However, rapid reduction of h after network condition resumes to a perfect condition can lead to instability or a collision.

Therefore, we opted to have the reduction of h value to be gradual, and in a front-to-back manner: the second vehicle in the platoon (the direct platoon leader's follower) initiates the reduction first, followed by the second vehicle and so on. This process allowed us to change the headway value without breaking string stability of the platoon. In addition, when network condition degrades, headway values can be increased in an opposite direction, starting from the platoon's tail to front. Equation 3.26 is used to determine a target h value when a loss occurs to increase the headway value, effectively increasing the separation between vehicles during less-than-ideal network condition. For every vehicle i, we increase the value of h to the newly computed h value gradually over a period of 2 seconds.

The algorithm we devised to reduce h value when the network is back to perfect condition is listed in Algorithm 1. We assume that every vehicle i knows its position within the platoon, which allows us to perform the reduction in a front-to-back manner. The inputs i, h, and δ correspond to the vehicle's index within the platoon, the current headway value and the decrement by which we reduce the headway value, respectively. Our idea relies on two concepts to minimize the change of instability: the reduction of headway value should be gradual, and the reduction should start at the front of the platoon. This idea can be achieved by exchanging extra information in the messages to achieve the desired outcome. However, to reduce the complexity, we simply use a node's index to determine the delay until the initiation of the gradual reduction process, and the duration of the gradual reduction process. The reduction algorithm is invoked periodically every 10 seconds when the network condition is perfect after a loss process.

Algorithm 1: Reduction of h when network condition is perfect

Input: i, h, δ
Output: Value of h is reducted gradually for vehicle i
Function `HeadwayReduction` (i, h, δ) **:**
    while *Network is perfect* **do**
        $ttu \leftarrow 2 \times i$ seconds
        $h_{\text{new}} \leftarrow \max\{h - \delta, L\}$
        $slope \leftarrow \dfrac{h_{\text{new}} - h}{ttu}$
        $delay \leftarrow ttu$
        call `LinearReduction`$(slope, h_{\text{new}}, delay)$

Function `LinearReduction` $(slope, h_{new}, delay)$ **:**
    Wait for $delay$ seconds
    while $h \neq h_{new}$ **do**
        $h \leftarrow h + 0.1slope$
        Wait for 100ms

Chapter 4

Performance Evaluation

4.1 Dynamic Time Headway Adaptation Results

As mentioned earlier, we run several experiments to decide on the lower and upper bounds for h since this is a necessary input to the dynamic headway assignment algorithms. The leader's acceleration profile shown in Table 4.1 and Figure 1 can achieve string stable platoon operation under no packet loss scenario using a minimum string stable headway value of 0.5. We also observed that a maximum headway value of 1.5 would result in string stable operation under network outage scenario, and therefore our adaptation algorithm uses a value of $h \in [0.5, 1.5]$.

Acceleration Profile (Sample Name)	**Simulation Time**	**Source**
Sample1 (`austin7`)	1918 seconds	Austin, TX
Sample2 (`atlanta1`)	4457 seconds	Atlanta, GA
Sample3 (`austin5`)	2614 seconds	Austin, TX
Sample4 (`austin3`)	2870 seconds	Austin, TX
Sample5 (`caltrans5`)	2400 seconds	CalTrans, California

Table 4.1: Acceleration profiles used for the experiments

Acceptable results must have string stability and no collision. Therefore, our simulations compare the performance of using the fixed minimum string stable headway for a given loss process and

acceleration profile against our dynamic adjustment approach. The minimum string stable value of h varies with acceleration profile, and the network loss process applied.

We used data obtained from NREL's website[1] that contains drive cycle data including acceleration, which is sampled every 1 second. Because our implementation of CACC relies on acceleration updates every 100ms, we used a simple linear interpolation to fill in the data points between every two acceleration to make it a per-100ms acceleration profile.

4.1.1 Results for 50/50 Loss Process

For each acceleration profile in Table 4.1, we run multiple CACC experiments with a constant headway time value h, and we applied a loss process of $MGL = 50, MBL = 50$ for 300 seconds (5 minutes), applied between simulation times 500 up to 800. The aim is to establish what is the minimum h that can be used without causing a collision or breaking string stability under the loss process. We then compare the traffic flow achieved by using that minimum string stable h with the adaptive approach as shown in Table 4.2, where traffic flow is quantified in units of *vehicles per hour*. In Figure 4.1, we observe that the overall traffic flow is increased for the simulation run with Sample1. The shaded area marks the time in which the loss process is applied. As we can see, the performance during the loss process matches the traffic flow of the scenario where the minimum string stable h was used. In Figures 4.3, we observe that during the loss process, Sample3's traffic flow is slightly decreased when compared to the fixed headway scenario. However, the overall achieved traffic flow is 15.54% higher, as seen in Table 4.2. The results from the remaining samples are shown in Figures 4.4, and 4.5.

Acceleration Profile	**Fixed h**	**Adaptive h**	**Improvement %**
Sample 1	2743.29	3385.45	+23.41%
Sample 2	2625.33	3284.89	+25.12%
Sample 3	2777.15	3208.74	+15.54%
Sample 4	2500.3	2807.37	+12.28%
Sample 5	2869.26	3286.39	+14.54%

Table 4.2: Traffic throughput (*vehicle/hour*) achieved under a 50/50 loss process with different acceleration profiles.

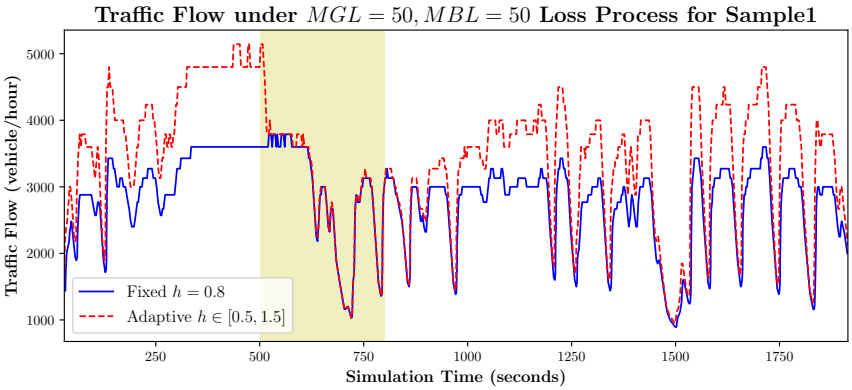


Figure 4.1: Sample 1 flow comparison under 50/50 process

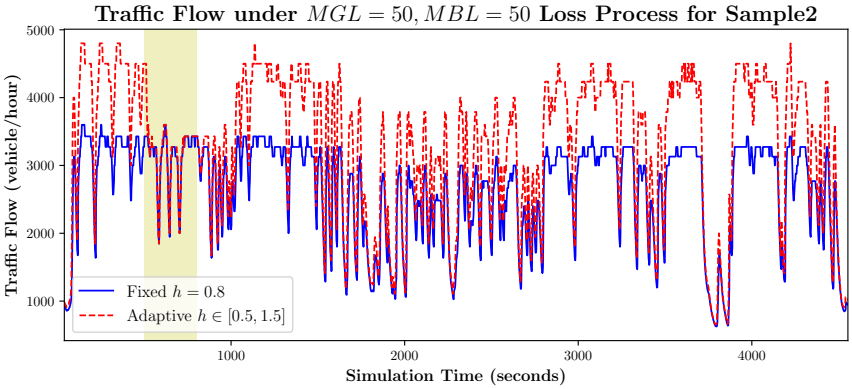


Figure 4.2: Sample 2 flow comparison under 50/50 process

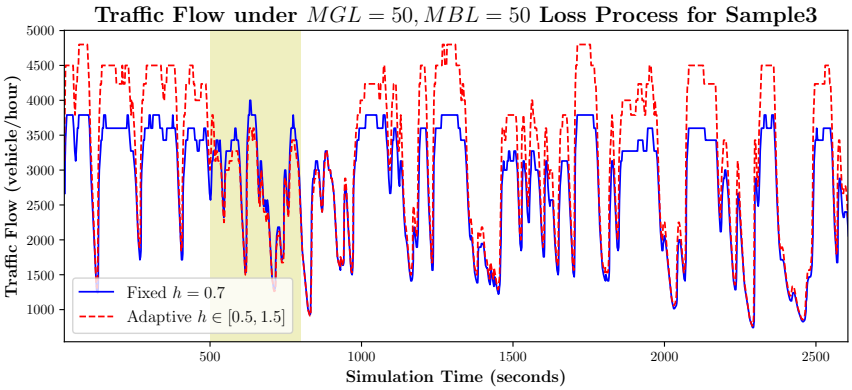


Figure 4.3: Sample 3 flow comparison under 50/50 process

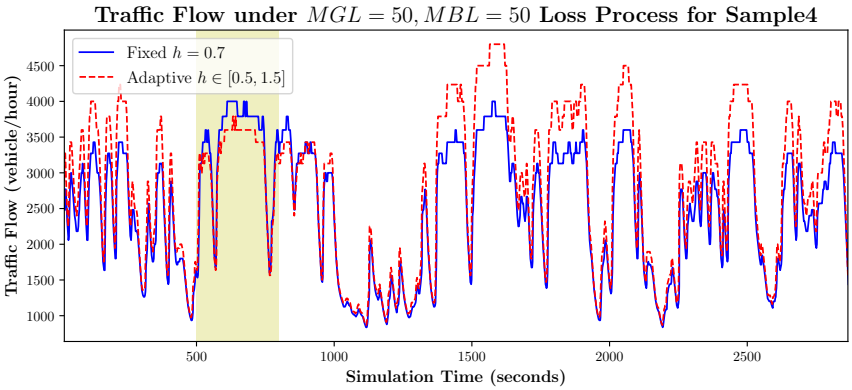


Figure 4.4: Sample 4 flow comparison under 50/50 process

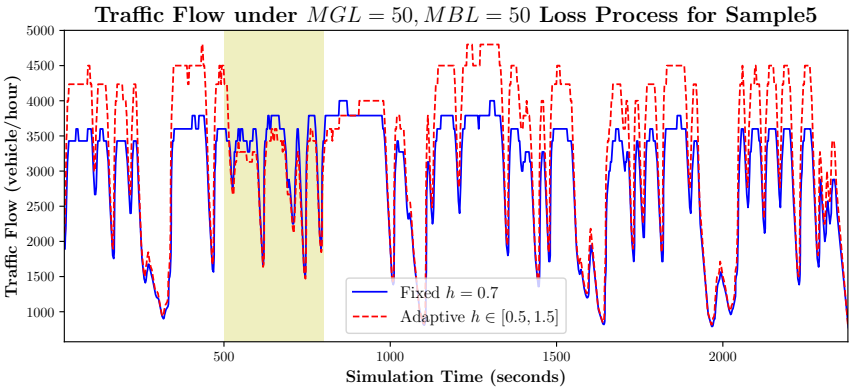


Figure 4.5: Sample 5 flow comparison under 50/50 process

4.1.2 Network Outage

We tested our proposed adaptation on a scenario where total network outage occurs. Our early tests established that a minimum headway of $h = 1.5$ is string stable under a total network outage scenario for the acceleration profiles we used. Therefore, in the case of a total network outage, the adaptation algorithm increases the headway to 1.5 during the loss process. Under this loss process, samples that can maintain string stable operation at fixed headway values $h < 1.5$ will have higher traffic flow during loss since total network loss always sets the headway to 1.5 as can be seen in Figures 4.9 and Figures 4.10. In addition, recovering from a total network outage scenario is relatively slow over time, but the adaptive algorithm gradually increases the traffic flow after the network resumes, where it matches the fixed algorithm at about the 1100th seconds mark, and then increases the traffic flow afterwards.

Acceleration Profile	**Fixed h**	**Adaptive h**	**Improvement %**
Sample 1	1813.95	2621.93	+44.54%
Sample 2	1761.46	2916.65	+65.58%
Sample 3	1737.8	2652.07	+52.61%
Sample 4	1680.66	2374.74	+41.3%
Sample 5	1953.8	2588.77	+32.5%

Table 4.3: Traffic throughput (*vehicle/hour*) achieved under a network outage scenario with different acceleration profiles.

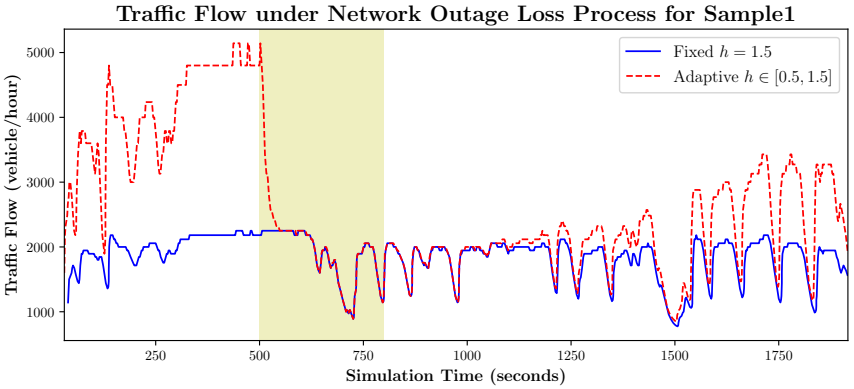


Figure 4.6: Sample 1 flow comparison under network outage

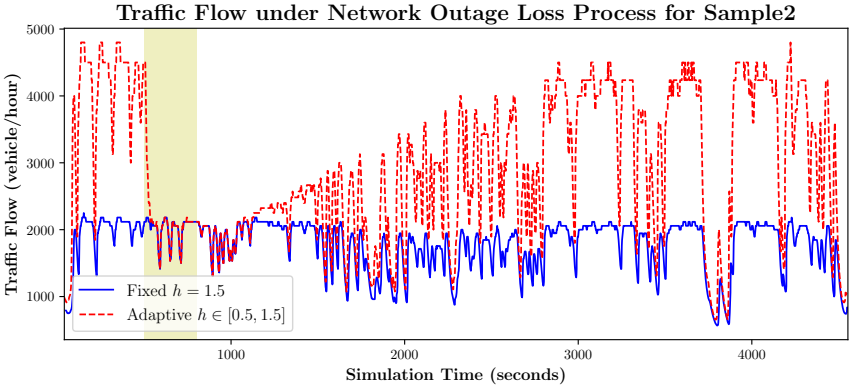


Figure 4.7: Sample 2 flow comparison under network outage

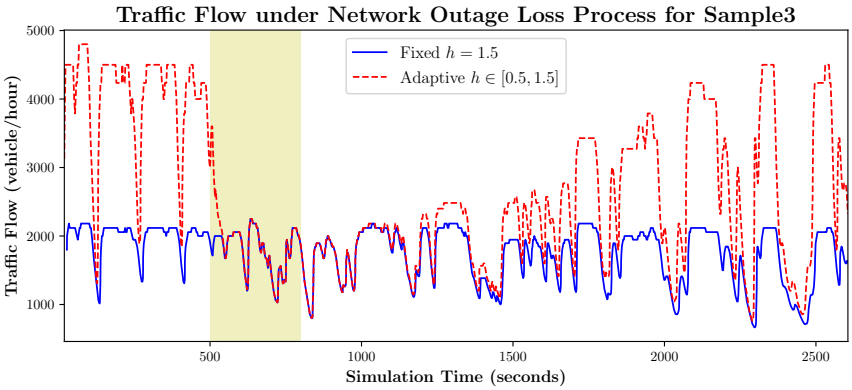


Figure 4.8: Sample 3 flow comparison under network outage

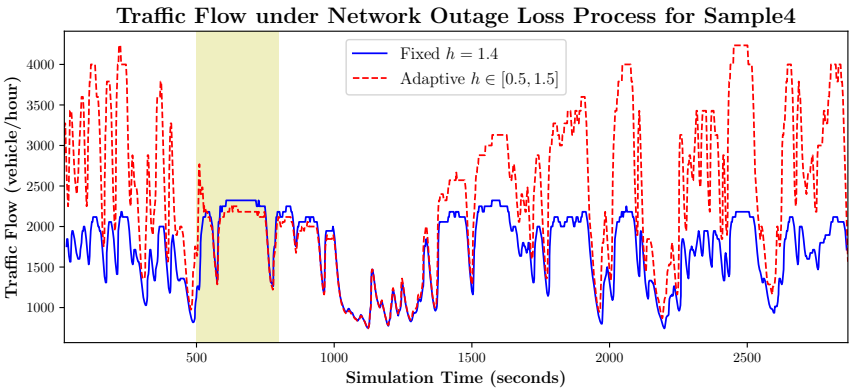


Figure 4.9: Sample 4 flow comparison under network outage

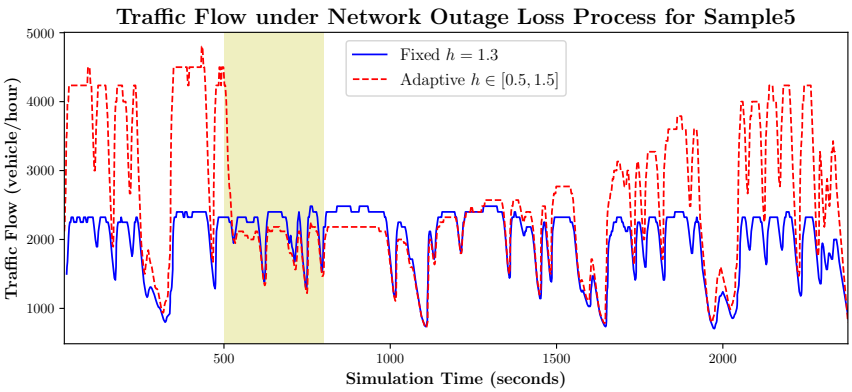


Figure 4.10: Sample 5 flow comparison under network outage

4.1.3 Congested Network : $MGL = 10, MBL = 40$

We applied our approach to a network with a loss process of $MGL = 10, MBL = 40$, and we noticed that our platoon adapted to the loss process well as seen in Figure 4.11, 4.12 and 4.13. The remaining samples set h slightly higher than the minimum string stable h, as our proposed solution is cautious when evaluation network quality. In the previous experiments with network outage, the adaptation algorithm assumes that the network is at its worse condition, and therefore, sets the h value to 1.5, its maximum value. We can apply that method to all loss processes on the expense of reduced traffic flow during loss. Therefore, the design of a platoon controller with adaptive h value may need to have a trade-off between being too cautious when increasing h value due to loss on the expense of reducing traffic flow during the loss event. It is also possible to incorporate factors such as degree of acceleration jerk and change to adapt the h value as well, but that is beyond the scope of this work.

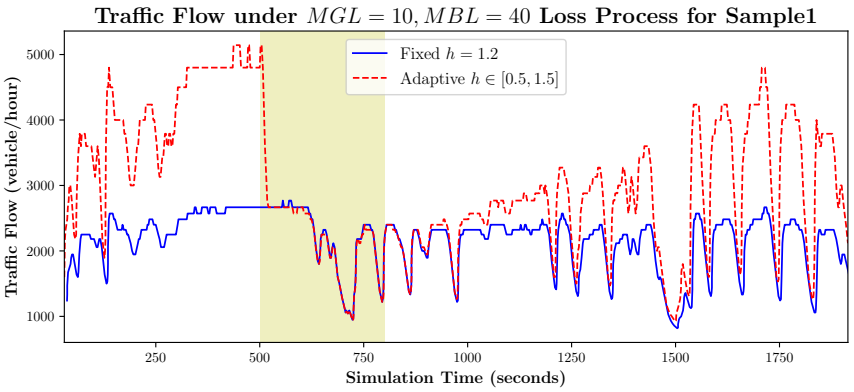


Figure 4.11: Sample 1 flow comparison under congestion

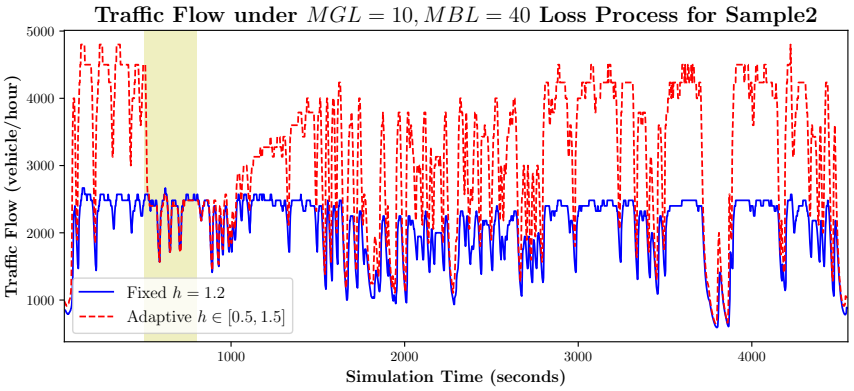


Figure 4.12: Sample 2 flow comparison under congestion

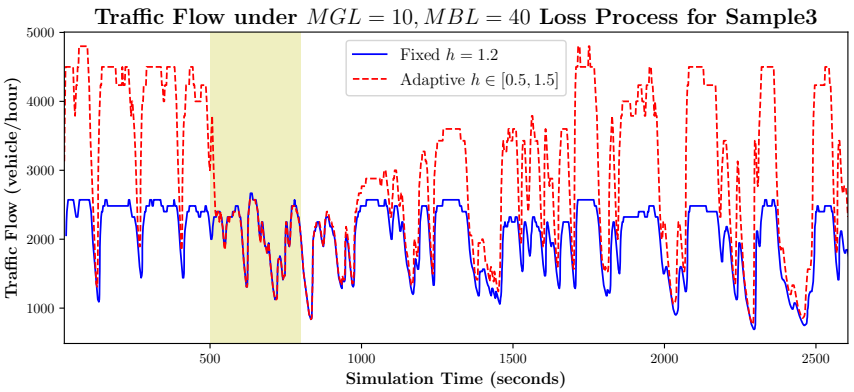


Figure 4.13: Sample 3 flow comparison under congestion

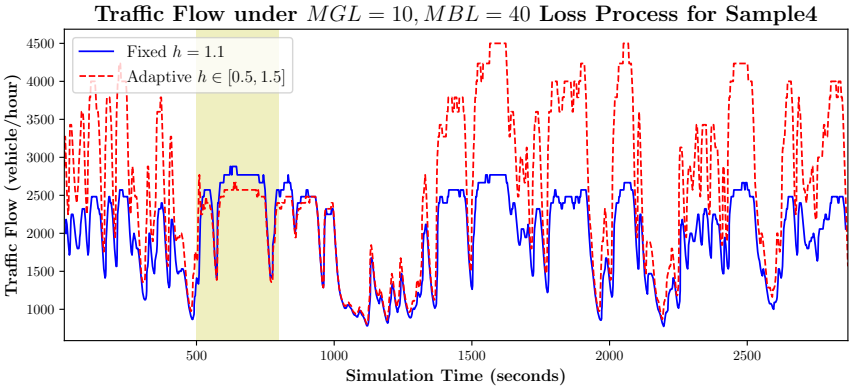


Figure 4.14: Sample 4 flow comparison under congestion

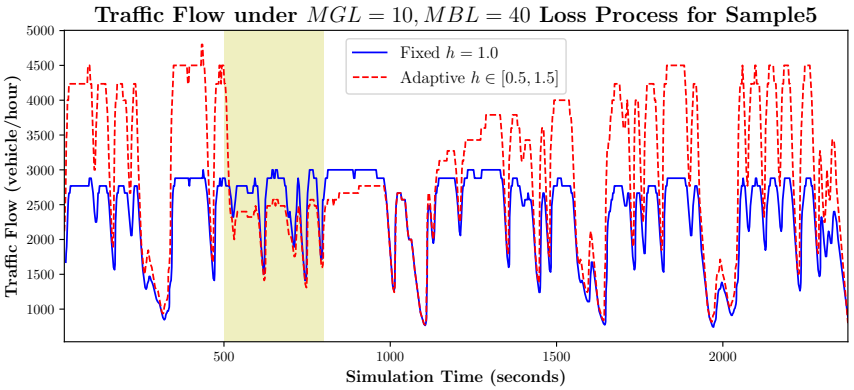


Figure 4.15: Sample 5 flow comparison under congestion

Acceleration Profile	**Fixed h**	**Adaptive h**	**Improvement %**
Sample 1	2121.77	2958.2	+39.42%
Sample 2	2046.14	3077.55	+50.41%
Sample 3	2018.25	2917.44	+44.55%
Sample 4	1946.37	2593.83	+33.27%
Sample 5	2318.36	2906.6	+25.37%

Table 4.4: Traffic throughput (*vehicle/hour*) achieved under a network congestion with different acceleration profiles.

4.2 Evaluation of 5G & DSRC in CACC

The following are results from comparing a CACC performance evaluation using 802.11-based DSRC and 5G NR V2X. We run our CACC simulation where vehicles are equipped with a DSRC/WAVE network device, as implemented in ns-3's built-in `wave` module. We ran CACC simulations with 10 vehicles to find the minimum string stable time headway for the given leader acceleration profile that satisfies Equation 3.9. Furthermore, we vary the value of h by 0.1 seconds with each experiment, and we only consider simulations that are string stable. Each vehicle broadcasts its acceleration information every 100ms. Similarly, for 5G NR V2X, we used the `5G-LENA` extension to ns-3 [10], which implements 5G V2X sidelink communication between vehicles, and uses the autonomous mode-2 for resource allocation to use the wireless sidelink channel. For both scenarios, vehicles only use acceleration value received from the direct preceding vehicle to calculate their new target acceleration. We also randomized the starting time for the applications installed in vehicles to reduce the probability of broadcasting their data at the same time.

In Figure 4.16, we show traffic flow of experiments where the platoon leader follows a sinusoidal acceleration profile, which alternate between accelerating and braking. The platoon that uses 5G NR V2X can achieve slightly higher traffic flow over DSRC as it can use a time headway as low as 0.7 seconds compared to 0.8 seconds for DSRC.

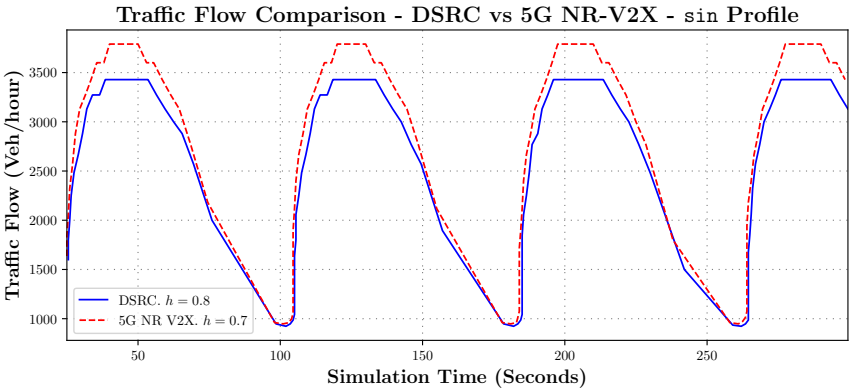


Figure 4.16: CACC Performance using DSRC versus 5G NR V2X - Sinusoidal Profile

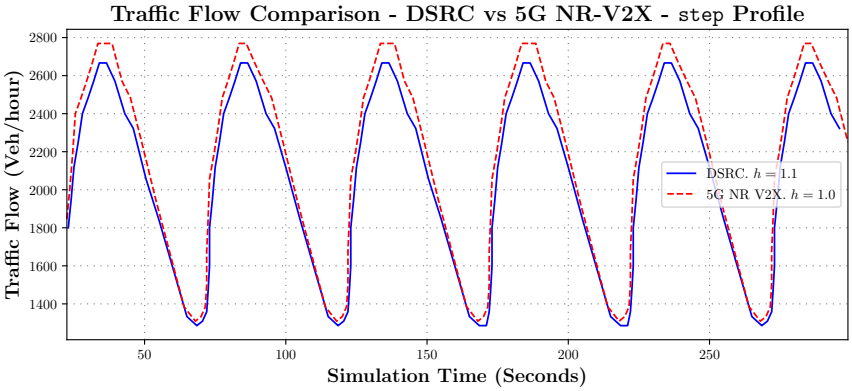


Figure 4.17: CACC Performance using DSRC versus 5G NR V2X - Step Profile

In Figure 4.17, we show traffic flow comparison on experiments where the platoon leader uses an acceleration profile that follows a step function, meaning that it would have sudden shifts between acceleration and braking. Again, the platoon that uses 5G NR V2X achieved slightly higher traffic throughput as it can operate at slightly lower time headway than the platoon using DSRC. We also observe similar results with the `us06` drive cycle profile shown in Figure 4.18. Figure 4.19 illustrates the results when using the *c2a* acceleration profile, showing significant performance advantage for 5G NR V2X over DSRC, as it can operate at a minimum string stable headway of $h = 0.7$, compared to $h = 1.1$ for DSRC.

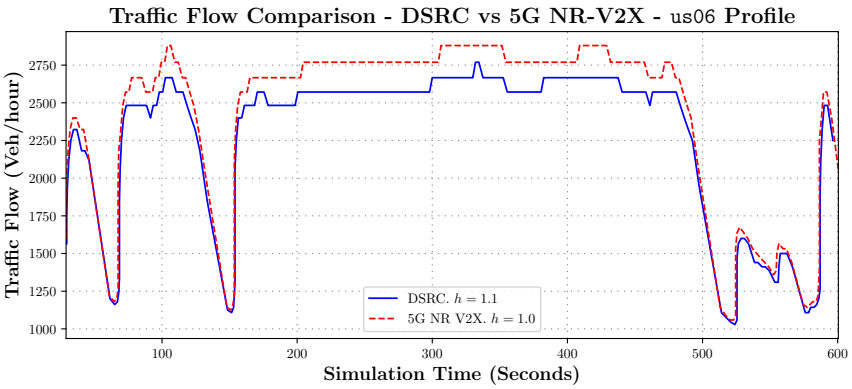


Figure 4.18: CACC Performance using DSRC versus 5G NR V2X - us06 Profile

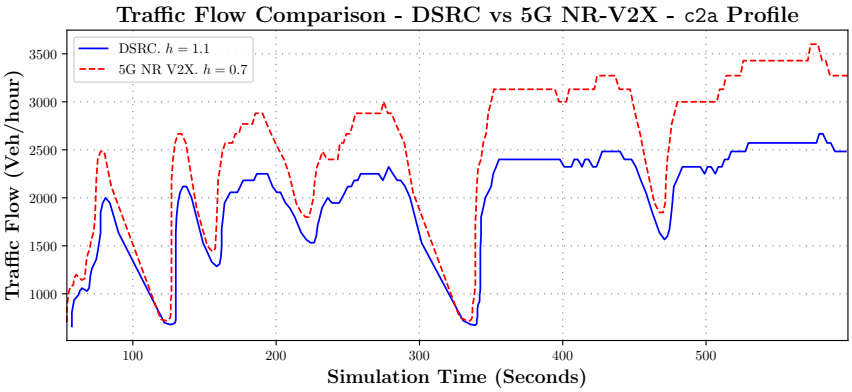


Figure 4.19: CACC Performance using DSRC versus 5G NR V2X - c2a Profile

We also used acceleration profiles from real vehicles that we obtained from NREL, and for sake of improving simulation speed, we only considered the first 600 seconds of each sample. We can see in Figure 4.20, 4.21 and 4.24 that using 5G NR V2X communication achieves significantly better traffic throughput as we can operate a string stable platoon at lower time headway values of h. The samples in Figures 4.22 and 4.23 show that 5G NR V2X have a slight advantage over DSRC.

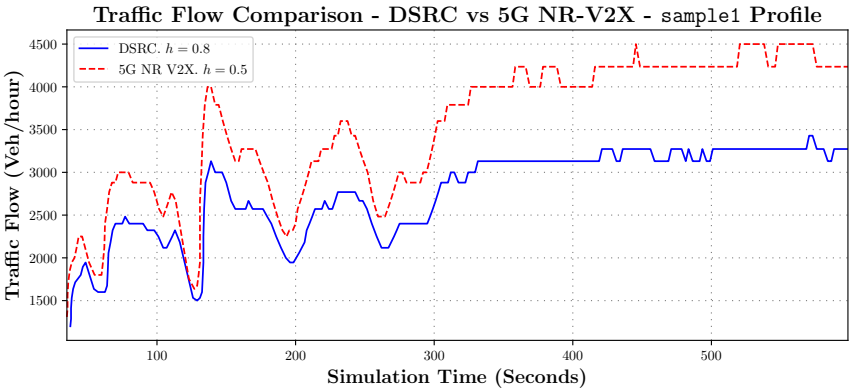


Figure 4.20: CACC Performance using DSRC versus 5G NR V2X - `Sample 1` Profile

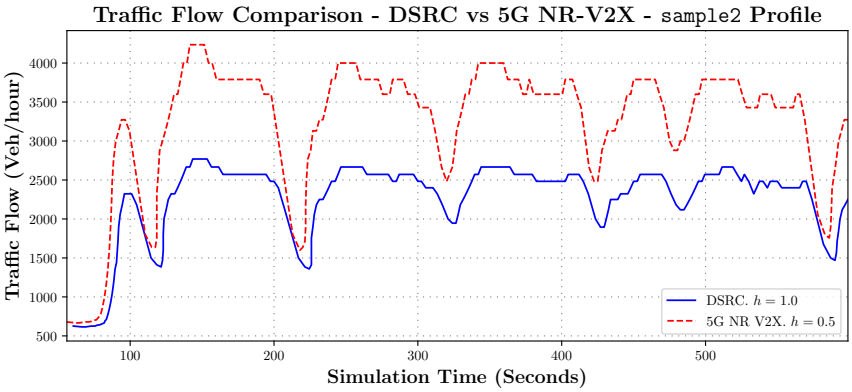


Figure 4.21: CACC Performance using DSRC versus 5G NR V2X - `Sample 2` Profile

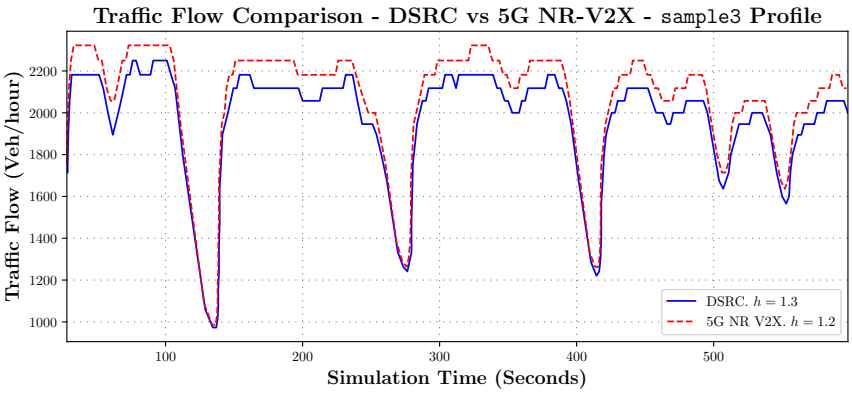


Figure 4.22: CACC Performance using DSRC versus 5G NR V2X - `Sample 3` Profile

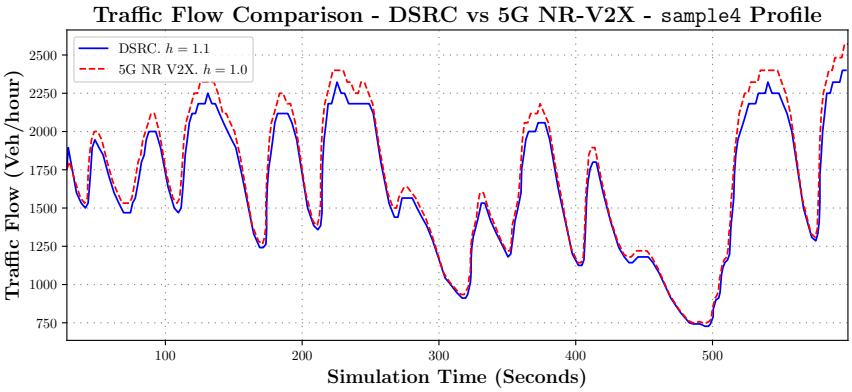


Figure 4.23: CACC Performance using DSRC versus 5G NR V2X - `Sample 4` Profile

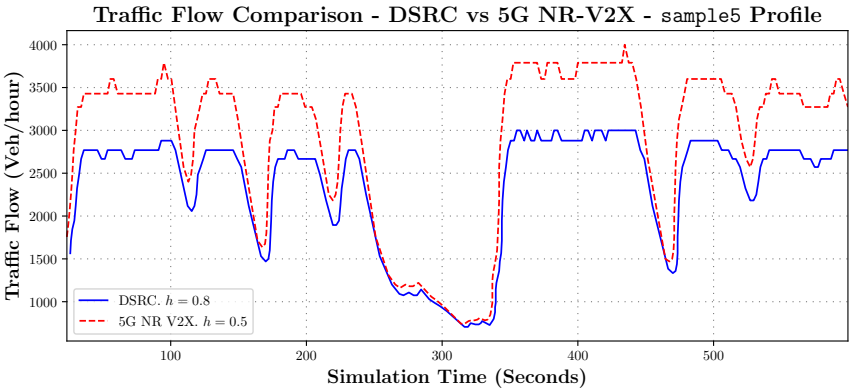


Figure 4.24: CACC Performance using DSRC versus 5G NR V2X - `Sample 5` Profile

We can attribute the performance variation to the fact that DSRC nodes use a relatively small backoff counter when the channel is sensed as "busy". The smaller value was implemented to improve the latency of DSRC applications, but we believe, based on our simulation results, that it comes at the expense of better packet delivery ratio as it would increase the chance of two or more nodes attempting to transmit at the same time. In addition, since applications' start up times and backoff numbers are random, we would expect varying degrees of performance, and it seems that 5G NR V2X handles that better than DSRC. DSRC can enter into a severe loss state based that it does not react to, leading to degraded performance of VANET applications such as CACC.

4.3 Evaluation of Wireless Technologies in Periodic Broadcast Applications

We implemented ns-3 simulations to investigate packet delivery ratio (PDR) achieved with applications that periodically broadcast 500-byte messages every 100ms. We ran experiments with varying number of nodes to see the effect of congestion on PDR as it could potentially leads to collisions. The vehicles are placed in multiple lanes to simulate how they would be in a real-world scenario. For DSRC, we used 10 MHz and 20 MHz channels that are supported by the original 802.11p amendment, and the future 802.11bd amendment (Enhancement for Next-Generation V2X, or NGV), which introduces modern modulation/coding schemes to VANET applications. C-V2X supports channel widths of 10 MHz, 20 MHz, 30 MHz and 40 MHz, but we only compare the 10 MHz and 20 MHz channels operating in the 5.9 GHz band to have a good comparison between the performance of C-V2X and DSRC.

4.3.1 DSRC and 802.11 PHY Modes

We ran ns-3 simulations with nodes equipped with 802.11-based network devices that are operating in OCB mode. We find the PDR that is achieved with the given number of nodes using different modulation/coding scheme (MCS). We repeat the experiments but with more nodes to study the effect of congestion on PDR. Using higher order of MCS means that a wireless frame's transmission and reception is completed in less time since the MCS supports a relatively higher data rate. We show the PDR from our experiments in Figure 4.25 using the highest MCSs supported by the 10 MHz channels of 802.11p and the highest MCSs that would be supported by the upcoming 802.11bd amendment in 20 MHz channels.

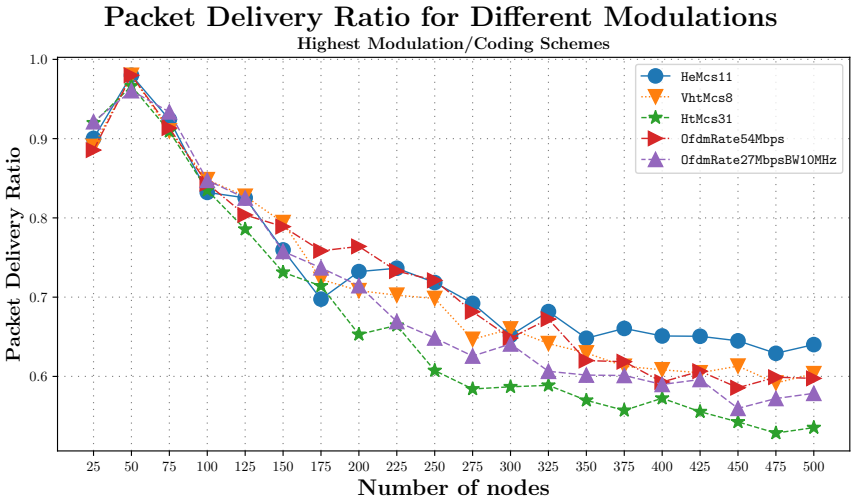


Figure 4.25: Packet Delivery Ratio for 802.11-based 10 MHz & 20 MHz channels - Highest MCSs

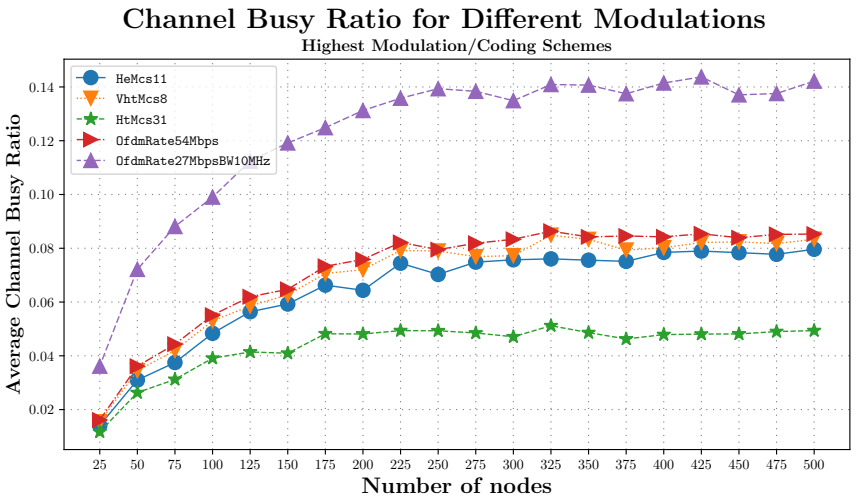


Figure 4.26: Average Channel Busy Ratio for 802.11-based 10 MHz & 20 MHz channels - Highest MCSs

Using higher order of MCS translates to an increase in effective data rate used to transmit a frame, and therefore, lower Channel Busy Ratio (CBR). CBR is the ratio of the channel's non-idle time to the total time. We expect that the lower channel usage time achieved with higher MCS means fewer frame collisions since the channel would be used for less time. The highest MCS supported by DSRC, which is named `OfdmaRate27MbpsBW10MHz` mode in ns-3, achieves a low CBR of 14% in the case of a congested network with 500 vehicles within range. Other modulations supported by

802.11 in 20 MHz channels can achieve even lower CBR as shown in Figure 4.26, with `HtMcs31`, supported by 802.11n and newer, having less than 5% CBR in a congested network. However, it only achieves about 53% packet delivery ratio as shown in 4.25. These lower values of CBR mean that higher orders of modulation/coding schemes result in less utilization of channel time.

However, our results in Figure 4.28 show that the MCS, `OfdmaRate3MbpsBW10MHz`, which is supported by DSRC, achieved the highest PDR of all other MCSs, with PDR value over 85%, but a congested network with a CBR of about 70%. We can attribute that to the fact that the longer channel usage time may introduce additional backoffs as nodes become more aware of other nodes' presence due to the extended channel usage times.

Figure 4.27 shows that the lowest MCSs of each 802.11 modes results in CBR values exceeding 60%, where we notice that CBR achieved by future modulations offer similar average CBR to the modulations currently supported by DSRC. The rate of CBR becomes flat in the plots due to the experiment setup which places the nodes in three long lanes of vehicle because the effect of the distance becomes relevant to both CBR and PDR.

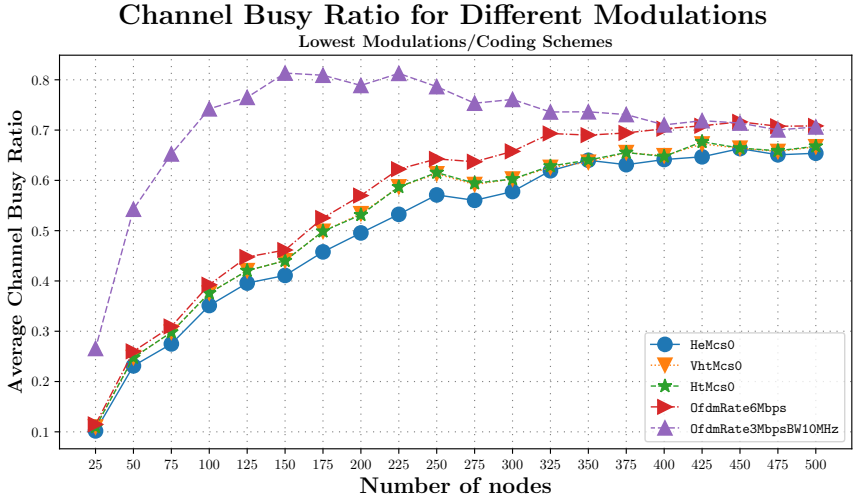


Figure 4.27: Average Channel Busy Ratio for 802.11-based 10 MHz & 20 MHz channels - Lowest MCSs

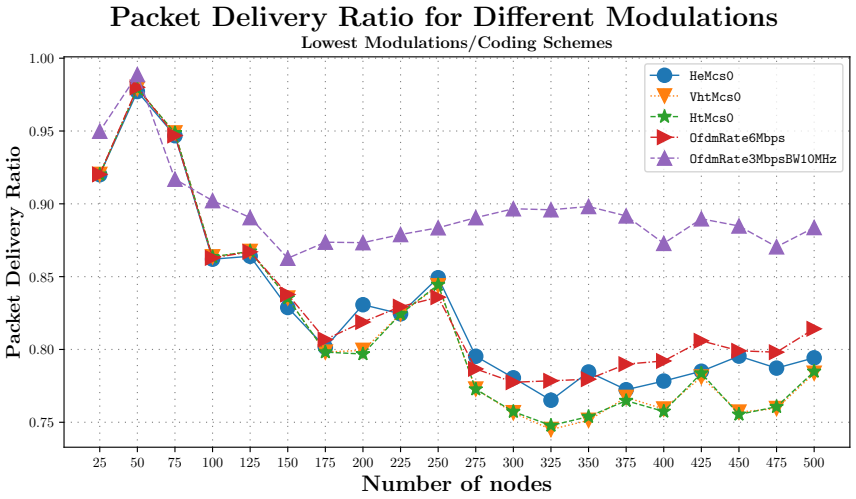


Figure 4.28: Packet Delivery Ratio for 802.11-based 10 MHz & 20 MHz channels - Lowest MCSs

The previous results were run by choosing the highest priority QoS traffic identifier (TID) as shown in Table 2.2. These access categories correspond to different backoff values generated during the distributed channel access process, so we ran another set of experiments using all four categories and the default MCS used in DSRC that corresponds to 6 Mbps data rate.

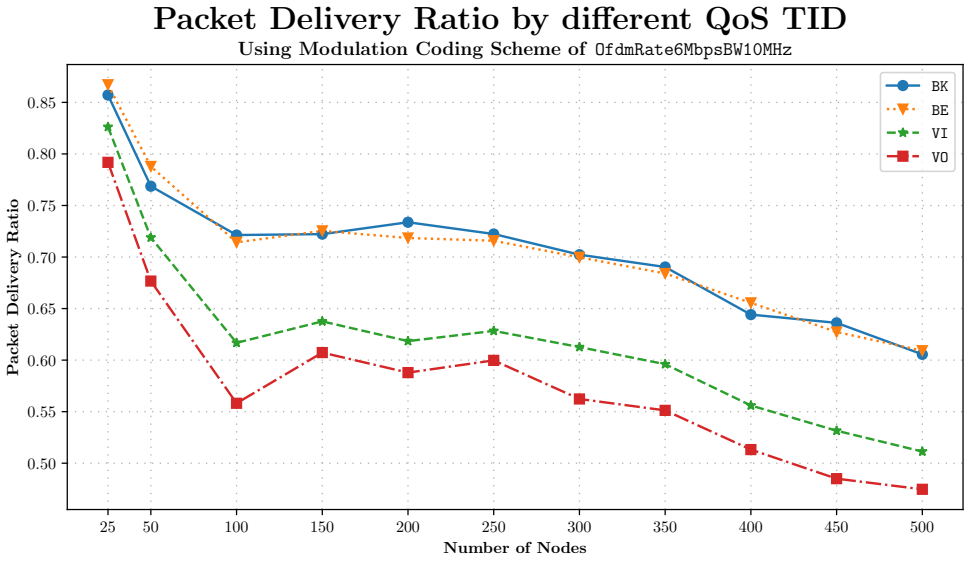


Figure 4.29: PDR Comparison of Different Access Categories using WAVE's `OfdmRate6MbpsBW10MHz` MCS

We observe in Figure 4.29 that generating larger backoff times in the cases of `BE` and `BK` access

categories that it is possible to achieve higher packet delivery ratio due to lower number of collisions.

4.3.2 C-V2X based on 5G NR

We obtained an extension to ns-3 that supports 5G NR communication from CTTC (in Barcelona, Spain), which also includes an extension to support sidelink V2X communication. We implemented simulations that follow the same procedure for DSRC, where we find the PDR achieved with increasing number of nodes.

We ran simulations where vehicles are equipped with 5G NR V2X devices, and we varied the number of vehicles with each experiment up to a heavily congested network of 250 vehicles within proximity of each other. Our results, shown in Figure 4.30, show that for some MCS, we can achieve packet delivery ratios that are higher than 90% when using a 10 MHz sidelink channel, which outperforms the 802.11 setup that uses 10 and 20 MHz channels with the same level of congestion. Figure 4.31 shows the results when using a 20 MHz sidelink channel, achieving PDR greater than 95% for some MCSs when the network is congested with 250 vehicles.

It is evident that the MAC protocol in 5G NR V2X, which is based on sensing-based semi-persistent scheduling, scales better than DSRC for congested networks of periodic VANET applications. We showed in Section 4.2 that using that mechanism in CACC applications can result in improved traffic flow as it would allow for smaller separation between a platoon's vehicles.

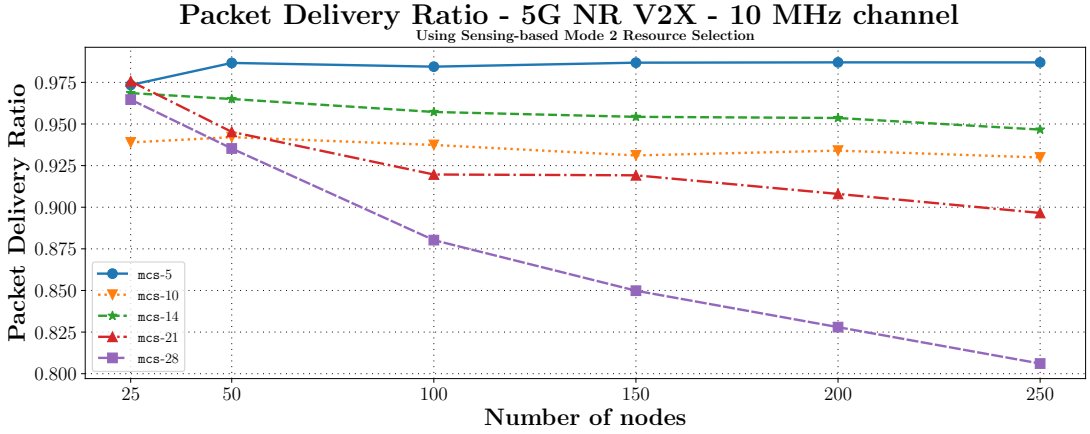


Figure 4.30: Packet Delivery Ratio 5G NR V2X-based 10 MHz Channel - Different MCSs

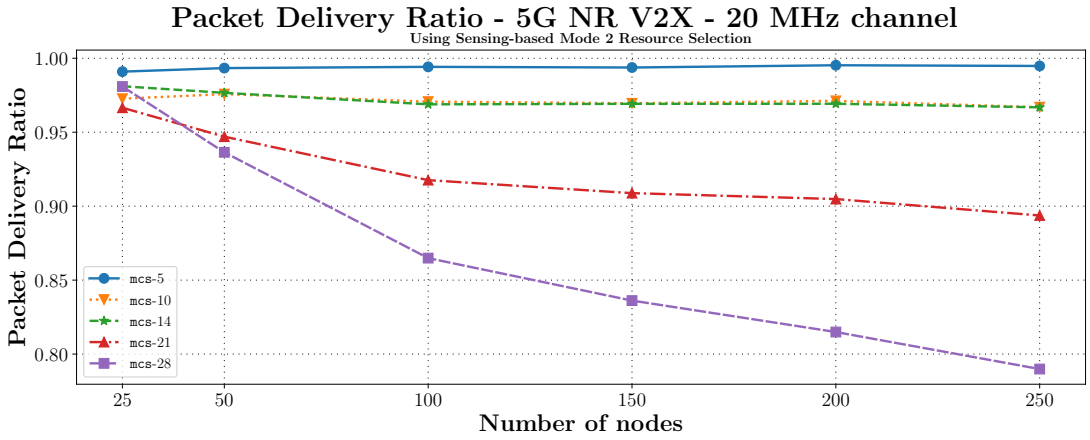


Figure 4.31: Packet Delivery Ratio 5G NR V2X-based 20 MHz Channel - Different MCSs

Chapter 5

Proposed Enhancements & Suggestions

5.1 Improved Kalman Filter for Acceleration Estimation

In our CACC experiments, a vehicle uses on-board sensors to estimate the acceleration of the preceding vehicle in the absence of the wireless network. The Kalman filter that is based on [73] is used to improve the estimation of a moving vehicle's acceleration and distance, but it relies on two independent sensors for distance and relative velocity. The estimated acceleration suffers from what is known as phase lag error as illustrated in Figure 5.1. In this section we introduce an improved Kalman filter that estimates the state of a preceding vehicle. We are not interested in the relative velocity of the preceding vehicle since the CACC model only needs acceleration and distance. The proposed Kalman filter can work with only a distance sensor, making it more cost-effective as it removes the need for two sensors. It assumes that the preceding vehicle being tracked has a constant-jerk, where jerk is the rate by which acceleration changes with respect to time.

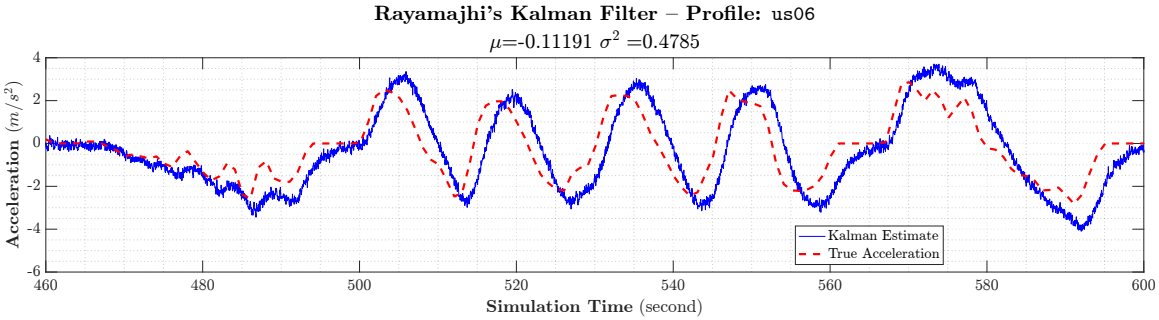


Figure 5.1: Lag Error in the current Kalman filter estimator. us06 Profile

5.1.1 Constant-Jerk Acceleration Estimator

The estimator that I developed for acceleration estimation uses a constant-jerk model, that is, the change in acceleration between two consecutive sensor reading is assumed to be constant. The 4-by-4 state transition matrix A is defined as:

$$A = \begin{bmatrix} 1 & \Delta t & \frac{(\Delta t)^2}{2} & \frac{(\Delta t)^3}{6} \\ 0 & 1 & \Delta t & \frac{(\Delta t)^2}{2} \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1}$$

The process noise covariance matrix Q is

$$Q = q \begin{bmatrix} \frac{\Delta t^7}{252} & \frac{\Delta t^6}{72} & \frac{\Delta t^5}{30} & \frac{\Delta t^4}{24} \\ \frac{\Delta t^6}{72} & \frac{\Delta t^5}{20} & \frac{\Delta t^4}{8} & \frac{\Delta t^3}{6} \\ \frac{\Delta t^5}{30} & \frac{\Delta t^4}{8} & \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} \\ \frac{\Delta t^4}{24} & \frac{\Delta t^3}{6} & \frac{\Delta t^2}{2} & \Delta t \end{bmatrix} \tag{5.2}$$

The measurement noise, R is a scalar defined as follows:

$$R = \frac{\frac{e_r^2}{4} + 2r}{3} \tag{5.3}$$

where e_r is the resolution of the digital encoder (sensor), r is $E[\epsilon_i]$ and ϵ_i is the associated error in measurement. The values are chosen for e_r to be 2.45 mm/count as per [72], and r is set to 0 because we generate Gaussian error with zero mean.

In this model, the observation matrix H is a 4-by-1 matrix with only the first element set to 1 because we assume that we can only observe the distance, and P is identity matrix because we assume that the first observation is correct

$$H = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.4}$$

The measurement z is a scalar denoting the current distance reading with added Gaussian noise. Notice that in this model, we are assuming that there is a single sensor that can measure distance from the preceding vehicle. The state of the system $\hat{x}[k]$ at time step k is a four-component vector for distance, relative velocity, relative acceleration and relative jerk of the preceding vehicle.

$$\hat{x}_k = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \end{bmatrix}, \text{ and initialized to } \hat{x}_0 = \begin{bmatrix} 3 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.5}$$

The B matrix and u (external input) are set to zeros because we do not consider external inputs in this model. Vehicles are initially separated by 3 meters and are stationary, and therefore, the initial state of the preceding vehicle is set to 3, 0, 0, 0. The Kalman filter takes as input $A, H, P, Q, \hat{x},$ and z and outputs new values for P, and new state $\hat{x}$ that contains the estimated distance, relative velocity, relative acceleration and relative jerk of preceding vehicle.

The Kalman filter's process is then outlined as follows:

$$\hat{x}_k \leftarrow A\hat{x}_{k-1} + Bu_{k-1} \quad \text{Predicted State Estimate} \tag{5.6}$$

$$P_k \leftarrow APA^T \quad \text{Predicted Error Covariance} \tag{5.7}$$

$$y \leftarrow z_k - H\hat{x}_k \quad \text{Innovation of Measurement pre-fit Residual} \tag{5.8}$$

$$S \leftarrow R + HPH^T \quad \text{Innovation of Pre-fit Residual Covariance} \tag{5.9}$$

$$K \leftarrow PH^T \times S^{-1} \quad \text{Optimal Kalman Gain} \tag{5.10}$$

$$\hat{x}_k \leftarrow \hat{x}_k + Ky \quad \text{Updated State Estimation} \tag{5.11}$$

$$P \leftarrow (I - KH)\, P \quad \text{Updated Estimation Covariance} \tag{5.12}$$

So after each sensor reading z_k at time step k, the Kalman filter updates the state variable $\hat{x}_k$ and estimation covariance matrix P. The components of the $\hat{x}$ vector are distance, relative velocity, relative acceleration and relative jerk of the preceding vehicle, and so to find the preceding vehicle's acceleration, vehicle $i-1$ adds its own acceleration to the value acquired from $\hat{x}$

5.1.2 Experiments and Analysis

We created a CACC simulation in ns-3 with two vehicles: a preceding vehicle that follows a preset acceleration profile, and a following vehicle that follows the CACC model to update its acceleration every 100ms based on acceleration information that it receives from the leader. At each sensor interval of 25ms, the following vehicle records the true distance, relative velocity and acceleration to the preceding vehicle using ns-3's mobility model, and also records the timestamp.

After a simulation concludes, we use the collected data in the proposed Kalman filter model to analyze its accuracy. The data reflect ideal sensor readings with no errors, and therefore, we add Gaussian noise with zero-mean and 0.1 m variance to every sensor reading to simulate real-world measurement noise, and pass the noisy measurement z to the Kalman filter to estimate acceleration based on the noisy measurement. The true values of the preceding vehicle's state is used to compare how well the Kalman filter performed in estimating the preceding its state

Figure 5.2 and 5.3 shows the improved performance when the preceding vehicle follows synthetic acceleration profile. The proposed estimator has improved accuracy in estimating the acceleration

of the preceding vehicle, whereas the previous estimator follows the shape of the acceleration profile, but is not very accurate as it has noticeable lag error.

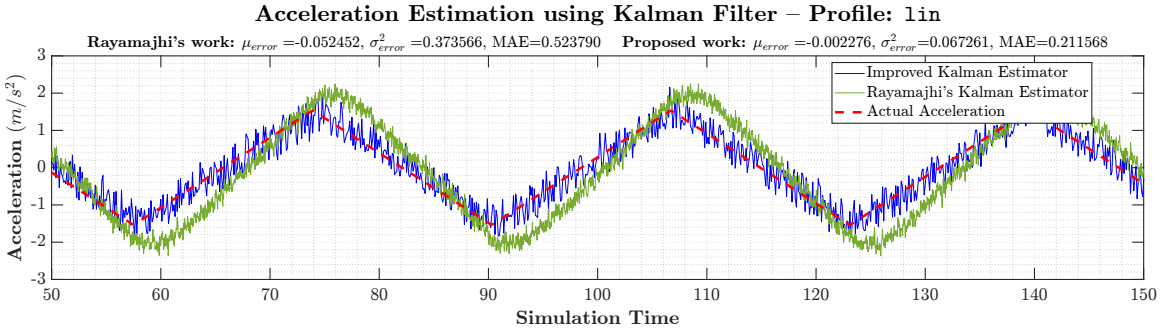


Figure 5.2: Improvement in acceleration estimation using linear acceleration profile

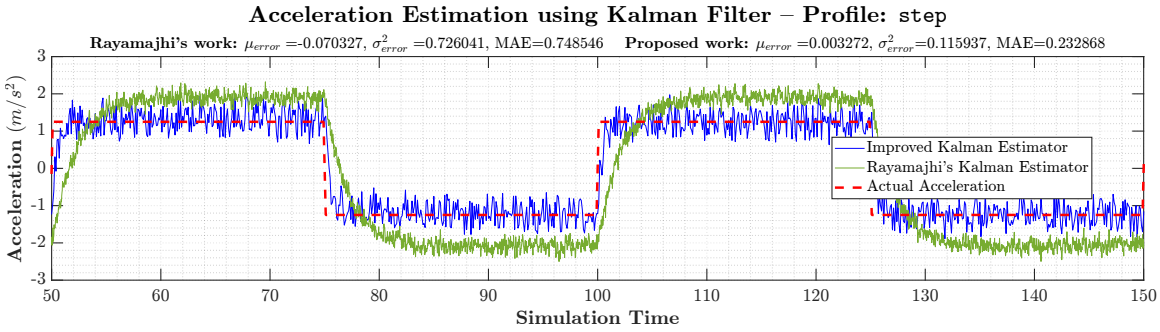


Figure 5.3: Improvement in acceleration estimation using linear step profile

Another way we can observe the improvement is by looking at the histograms of acceleration estimation errors shown in Figures 5.4 and 5.5 for linear and step profiles respectively. The improved Kalman filter shows significant improvement in acceleration estimation over the previously used filter.

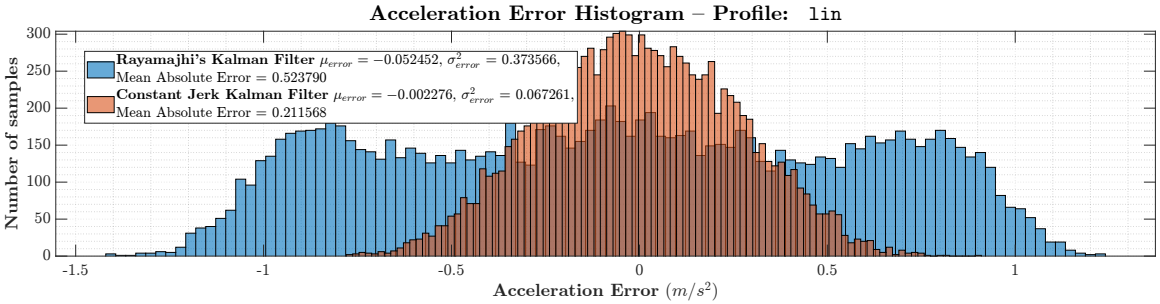


Figure 5.4: Histogram of acceleration estimation errors for linear acceleration profile

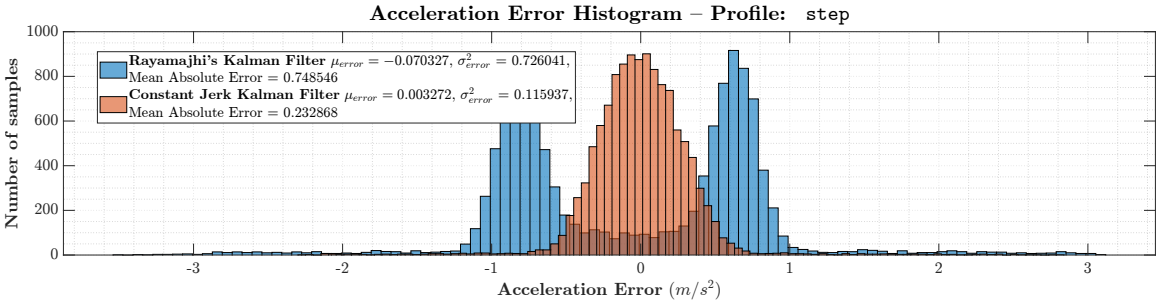


Figure 5.5: Histogram of acceleration estimation errors for step acceleration profile

We also tested with acceleration profiles obtained from real vehicles. We present a comparison of the estimations in Figure 5.6, and the histogram of the errors in Figure 5.7. The improvement in acceleration estimation is significant. We summarize the mean and variance of the error in acceleration estimation in Table 5.1. We observe that the improved constant-jerk Kalman filter consistently achieves lower mean error μ_{error}, and lower error variance σ_{error} for all the tested acceleration profiles. The remaining plots and histograms from the remaining acceleration profiles are shown in Appendix C.

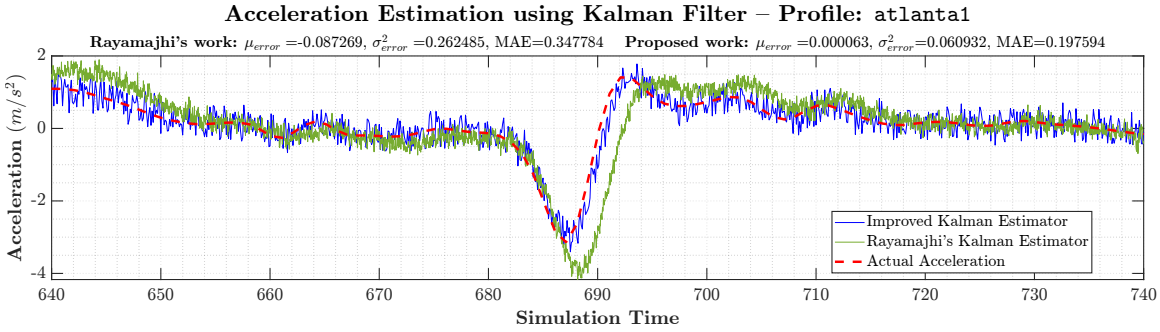


Figure 5.6: Improvement in acceleration estimation using `sample2` (`atlanta1`) acceleration profile

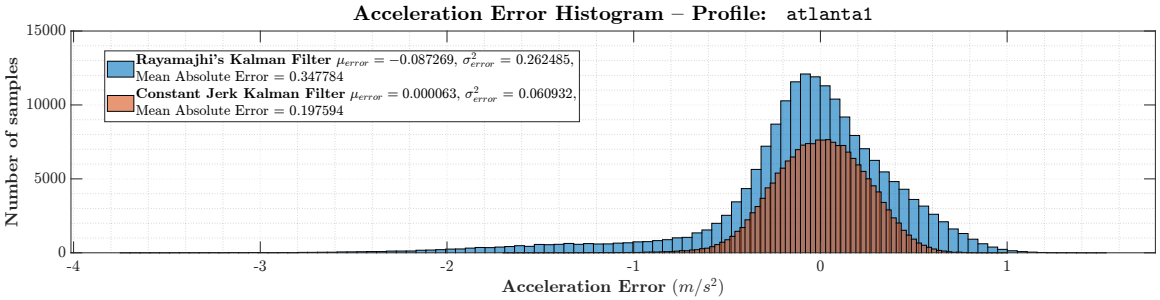


Figure 5.7: Histogram of acceleration estimation errors for `sample2` (`atlanta1`) acceleration profile

Sample	μ_{error} Rayamajhi	μ_{error} CJ	σ^2_{error} Rayamajhi	σ^2_{error} CJ
`lin`	-0.052452	-0.002276	0.373566	0.067261
`sin`	-0.075845	4.6E-05	0.379348	0.049086
`step`	-0.070327	0.003272	0.726041	0.115937
`c2a`	-0.085197	-0.000174	0.241007	0.095485
`us06`	-0.111913	-0.003165	0.481373	0.085707
`austin3`	-0.07636	-0.000198	0.213729	0.060402
`austin5`	-0.12228	-0.000112	0.293334	0.053131
`atlanta1`	-0.087269	6.3E-05	0.262485	0.060932
`caltrans5`	-0.091249	-0.001144	0.277307	0.058064
`austin7`	-0.094606	-0.000348	0.229802	0.057689

Table 5.1: Summary of mean and variance of acceleration estimation errors for all acceleration profiles

The CACC model also requires that vehicles estimate the distance from the preceding vehicle. The result from the improved Kalman filter estimator are compared to the previous filter for a synthetic linear profile, and an acceleration profile from a real vehicle in Figures 5.8 and 5.9 respectively. The improved Kalman filter achieves slightly better distance estimation than the previous work. In Figure 5.10, we notice the highest improvement in distance estimation because `step` acceleration profile exhibits a high degree of fluctuation between acceleration and deceleration. This means that the previously used Kalman estimator performs poorly when there are sudden changes in acceleration of the vehicle being tracked. Table 5.2 summarizes the error means and variance from distance estimation for all acceleration profiles tests. The remaining histograms for distance estimation are presented in Appendix D.

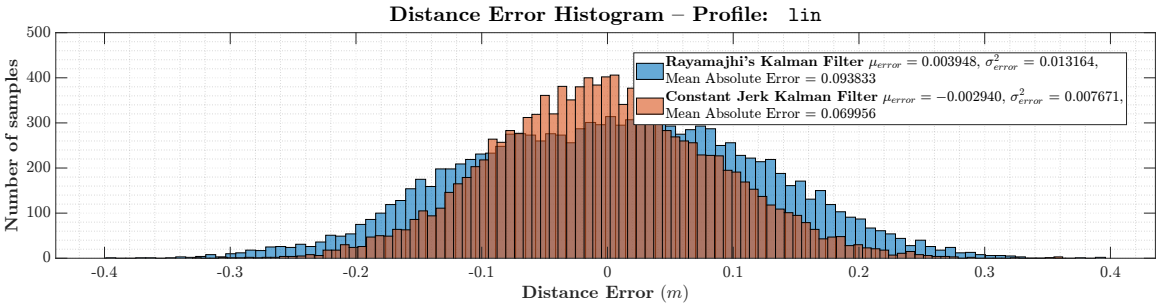


Figure 5.8: Distance error histogram. Linear profile

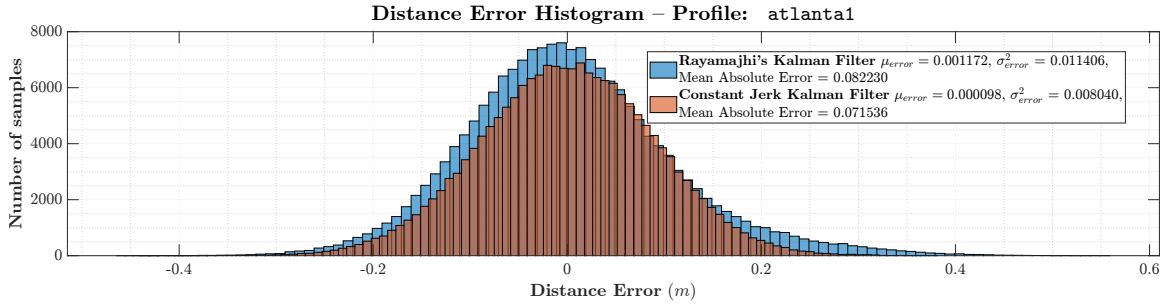


Figure 5.9: Distance error histogram. Sample 2 (`atlanta1`) profile

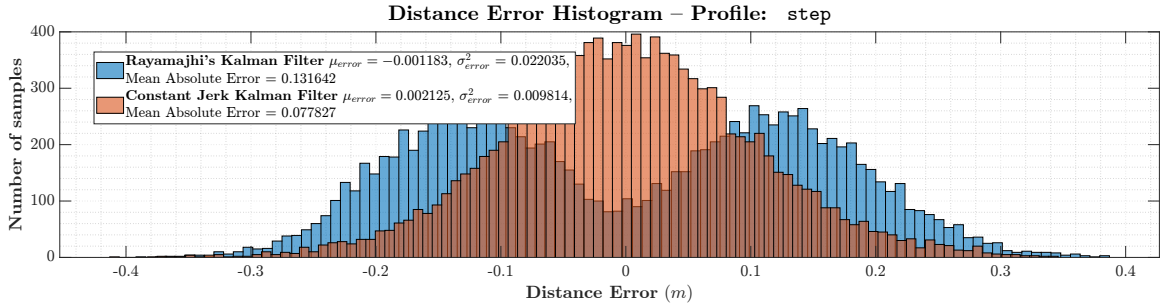


Figure 5.10: Distance error histogram. Step profile

Sample	μ_{error} **Rayamajhi**	μ_{error} **CJ**	σ^2_{error} **Rayamajhi**	σ^2_{error} **CJ**
`lin`	0.003948	-0.00294	0.013164	0.007671
`sin`	-0.004635	-0.001289	0.020845	0.007643
`step`	-0.001183	0.002125	0.022035	0.009814
`c2a`	-0.004366	4.2E-05	0.009221	0.008765
`us06`	-0.002796	0.000265	0.01396	0.008853
`austin3`	-1.7E-05	0.001694	0.009997	0.007954
`austin5`	-0.002715	0.001544	0.023263	0.008568
`atlanta1`	0.001172	9.8E-05	0.011406	0.00804
`caltrans5`	-0.001072	0.000469	0.01261	0.007892
`austin7`	-0.001067	0.000747	0.011027	0.007859

Table 5.2: Summary of mean and variance of distance estimation errors for all acceleration profiles

Since estimation errors can be positive and negative, the mean of estimation errors might not be very useful to give a clear picture of the accuracy of estimation. For a better analysis, we can look at the mean absolute error (MAE), which is the average of the absolute value of the errors in estimation, which would be more robust in this case as it measures the magnitude of deviation from the true value. Table 5.3 shows a summary of improvement in MAE in estimating distance and acceleration of the tracked vehicle for all acceleration profiles tested.

	MAE (Distance)		MAE (Acceleration)	
Sample	**Ryamajhi's work**	**Proposed Work**	**Ryamajhi's work**	**Proposed Work**
lin	0.093833	0.069956	0.52379	0.211568
sin	0.120335	0.070111	0.517802	0.181258
step	0.131642	0.077827	0.748546	0.232868
c2a	0.07533	0.074038	0.343471	0.238344
us06	0.089323	0.074794	0.485172	0.232425
austin3	0.078371	0.070916	0.328264	0.196697
austin5	0.115738	0.073803	0.397497	0.167546
atlanta1	0.08223	0.071536	0.347784	0.197594
caltrans5	0.087605	0.070718	0.380639	0.194582
austin7	0.081562	0.070689	0.345683	0.194339

Table 5.3: Mean Absolute Error (MAE) in estimation of distance (in meters) and acceleration (in m/s^2)

For distance estimation, we achieved improved estimation up to 0.0538 meters (5.38 cm) with an average improvement of 0.02316 meters (2.316 cm). For acceleration estimation, we achieve up to 0.515678 m/s^2 improvement in estimation with an average improvement of 0.2371 m/s^2.

5.2 DSRC+ : Employing Blind Retransmission in 802.11-based VANET

We observed in the previous experiments that simulations using 802.11-based radio performs poorly in case of congestion, with the most congested scenario of 500 vehicles resulting in packet delivery ratio that falls below 70%. We also observed that in such congested scenarios, the channel busy ratio (CBR) is below 20% for modern modulation/coding schemes that will be introduced with 802.11bd. ETSI's Technical Specification document for Decentralized Congestion Control in the 5 GHz band [84] suggests that vehicles detecting a CBR value below 30% would be in a *relaxed* state where no congestion control is necessary. We believe that the abundance of channel's idle time can be exploited by allowing nodes to perform blind retransmission of their frames.

5.2.1 DSRC+ in Periodic Broadcast VANET Applications

5.2.1.1 Experiment Setup

We implemented ns-3 simulations where we placed vehicles in multiple n lanes that have a longitudinal separation of Δx and lateral separation of Δy. The vehicles are stationary and broadcast

packets every 100ms, which is typical for VANET applications. The setup is illustrated in Figure 5.11.

We ran simulations for 120 seconds, tracking every single packet with a unique identification number to create a set of all transmitted packets S. When a packet is received by another vehicles, its unique identifier is added to the set of received packets R. We then computed *Packet Delivery Ratio* (PDR), as our evaluation metric as

$$PDR = \frac{|R|}{|S|} \tag{5.13}$$

For 802.11-based radio communication, we tested with all available modulation coding schemes listed in Table 5.4. All modulation/coding schemes use 20 MHz channels except for 802.11p, which uses 10 MHz channels.

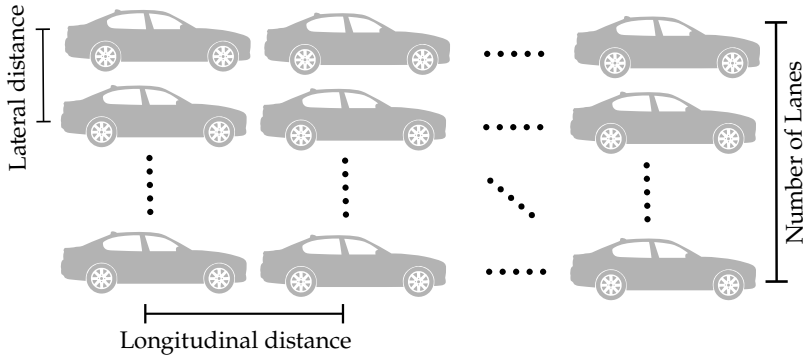


Figure 5.11: Vehicle Placement to Test Blind Re-Transmission

Standard	MCS ns-3	Modulation	Coding
802.11p	`OfdmRate3MbpsBW10MHz`	BPSK	1/2
	`OfdmRate27MbpsBW10MHz`	64-QAM	3/4
802.11a	`OfdmRate6Mbps`	BPSK	1/2
	`OfdmRate54Mbps`	64-QAM	3/4
802.11n	`HtMcs0`	BPSK	1/2
	`HtMcs31`	64-QAM	5/6
802.11ac	`VhtMcs0`	BPSK	1/2
	`VhtMcs8`	256-QAM	5/6
802.11ax	`HeMcs0`	BPSK	1/2
	`HeMcs11`	1024-QAM	5/6

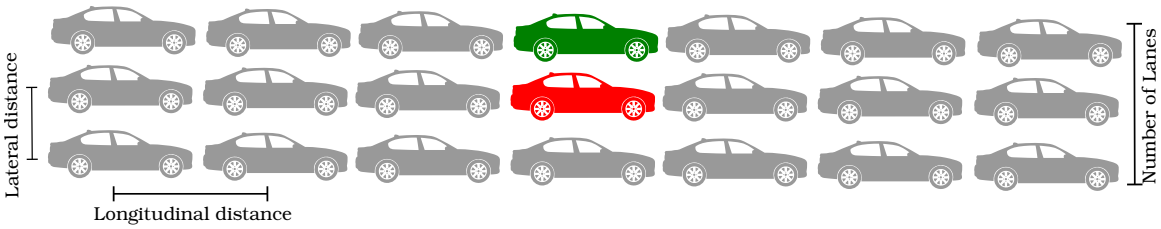
Table 5.4: 802.11 Modulation/Coding Schemes used in our experiments

In addition, the `wave` module in ns-3 allows us to track the channel's state, so we can find the average channel busy ratio. This is obtained by dividing the non-idle channel time by the total time.

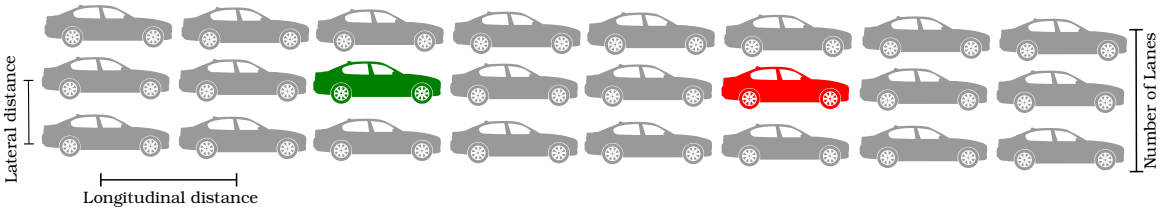
We also record the interarrival times of periodic broadcast messages between two nodes of interest. This is defined as the difference between two consecutive frames that are sent by node S and received by node R. We tested two scenarios for interarrival times:

1. **Adjacent Nodes:** The two vehicles of interest are in the exact center of the simulation environment, adjacent to one another, as illustrated in Figure 5.12a, where the vehicle in red is the sending vehicle S, and the vehicle in green is the receiving vehicle R.

2. **Separated Nodes:** The two vehicles of interest are separated from one another such as the sender vehicle S is at the end of the first third of the simulation setup, and the receiver vehicle R is at the end of the second third of the simulation setup as illustrated in Figure 5.12b.



(a) Adjacent Nodes for which Inter-arrival times are monitored



(b) Interest Nodes for monitoring inter-arrival times

Figure 5.12: Nodes of interest to monitor interarrival times

We ran two sets of experiments with different vehicle placement setups. In the first setup, vehicles were placed in 4 lanes, with a longitudinal separation (Δx) of 5 meters between vehicles, and a lateral separation (Δy) of 3.5 meters. This means that there are 125 rows of vehicles, giving us a distance of 625 meters between the first row and the last row. In the second setup, we created an 8-lane setup with longitudinal and lateral distance set to 1 meter. In this setup, the distance between the front row and the last row is 62.5, meaning that all nodes are within communication range of one another.

Finally, blind retransmission is performed using a probability that a frame would be retransmitted, and the values used are 10%, 25%, 30%, 50%, 75%, 80%, 90% and 100%. This is done by generating a random number between 0 and 1 drawn form a uniform random distribution, and if the number is smaller than or equal the probability, the frame would be retransmitted after a random delay between 0.050 ms and 90 ms, which means that the frame would be retransmitted within the same 100 ms interval. We also record the average One-Way Delay (OWD) for frames; we define OWD as the time between the generation of a packet at the application layer at a sending vehicle, to the time it is successfully received by other vehicles.

In addition to having a probability of retransmitting a wireless frame, we also experimented with scenarios where every frame is repeated more than once, that is, without generating a probability as in the previous setup. In these cases, an original frame transmitted at time t_{original} would be

transmitted an additional n times, where $n \in \{2, 3, 5\}$. The first additional frame is transmitted after t_{original} by a randomly-generated time between $50\mu s$ and $\frac{100}{n+1}$ms. The second additional frame is transmitted after the first additional frame by another randomly-generated time between $50\mu s$ and $\frac{100}{n+1}$ms, and so on, which ensures that all the additional frames are transmitted within the same 100 ms interval.

5.2.1.2 Results: Using 4 lanes, $\Delta x = 5, \Delta y = 3.5$ placement

The setup with $\Delta x = 5, \Delta y = 3.5$ is intended to mimic a real-world scenario of an interstate with 4 lanes and realistic separation distances. First, we ran experiments using the modulation/coding scheme supported by 802.11p (DSRC) in 10 MHz channel. We selected `OfdmRate12MbpsBW10MHz`, which corresponds to 16-QAM with coding rate of 1/2, and `OfdmRate27MbpsBW10MHz`, which corresponds to 64-QAM with coding rate of 3/4 (Refer to Table 2.1). Without using any repetition, the two nodes of interest do not have any interarrival information for the separated nodes scenario (as in Figure 5.12b), meaning that no packet was received by vehicle R from vehicle S. We observe that this is also the case with the faster MCS of `OfdmaRate27MbpsBW10MHz`, despite having a very low channel busy ratio (CBR) of about 9% as shown in Table 5.6.

AvgCBR	**PDR**	**AvgOWD**	**Avg. interarrival Separate Nodes**	**Avg. interarrival Adjacent Nodes**	**Repeat Probability**
0.13506	0.607539	0.00410995	∞	0.359847	0
0.276815	0.644611	0.00666209	1.31687	0.276391	0.1
0.461314	0.699098	0.00997724	0.550907	0.213556	0.25
0.507435	0.71605	0.0109988	0.458643	0.189486	0.3
0.605996	0.784127	0.0145062	0.45737	0.152783	0.5
0.640215	0.865796	0.0180104	0.54351	0.126357	0.75
0.64386	0.88023	0.018578	0.57167	0.123838	0.8
0.650991	0.910511	0.0197467	0.576602	0.116179	0.9
0.656655	0.941883	0.0207975	0.673203	0.108495	1

Table 5.5: Results from `OfdmaRate12MbpsBW10MHz` Modulation. 4 Lanes, $\Delta x = 5, \Delta y = 3.5$

Increasing the blind retransmission probability does increase the packet delivery ratio (PDR) for both MCSs. Moreover, interarrival times for the adjacent nodes scenario was lowered to about 100 ms. In addition, the interarrival times for the separated nodes scenario decreased from about 1.3 seconds for the case with retransmission probability of 0.1 as we increase the retransmission probability. At 1.0 probability, the packet delivery ratio exceeds 90%, but the average interarrival times for the separate nodes scenario remains relatively high at 306 ms for the `OfdmaRate27MbpsBW10MHz` MCS, and

673 ms for the `OfdmaRate12MbpsBW10MHz`.

The wireless channel also exhibit increased congestion, reaching 65% for `OfdmaRate12MbpsBW10MHz` when every frame is retransmitted. For modulation/coding schemes supported by the 802.11p standard, the blind retransmission mechanism does yield increased packet delivery ratio, but on the expense of causing congestion in the network.

Avg. CBR	**PDR**	**AvgOWD**	**Avg. interarrival Separate Nodes**	**Avg. interarrival Adjacent Nodes**	**Repeat Probability**
0.0895888	0.620554	0.00237223	∞	0.608286	0
0.165444	0.656604	0.00485156	1.28824	0.34617	0.1
0.271255	0.70994	0.00820364	0.486481	0.23237	0.25
0.302065	0.724234	0.00915855	0.409117	0.224098	0.3
0.397983	0.795306	0.0126221	0.325032	0.161695	0.5
0.4561	0.878825	0.0161305	0.278435	0.129628	0.75
0.461914	0.895442	0.0166856	0.28086	0.122649	0.8
0.472696	0.927713	0.0178681	0.264662	0.118468	0.9
0.480296	0.958962	0.0188388	0.306099	0.11198	1

Table 5.6: Results from `OfdmaRate27MbpsBW10MHz` Modulation. 4 Lanes, $\Delta x = 5, \Delta y = 3.5$

IEEE 802.11a-1999 supports 20 MHz channels that offer double the data rates of the corresponding 10 MHz channel. We use the highest MCS that corresponds to 54 Mbps, which is called `OfdmRate54Mbps` in ns-3. In these experiments, the results show the significant improvement that the 20 MHz channel offers over the 10 MHz channel width used in 802.11p. The results in Table 5.7 show that with a retransmission probability of 0.5, the PDR increases by about 18% from the experiments without blind retransmission, while maintaining a relatively low channel busy ratio of 0.25. Increasing the probability to 1 results in 0.9765 PDR, with a slightly congested network, as CBR reaches 0.38. In addition, the interarrival times for both scenarios is reduced, reaching 132.8 ms and 83.34 ms for the separate and adjacent node scenarios respectively.

We also tested with modulation/coding schemes offered by 802.11n, 802.11ac, and 802.11ax. The best results observed where from the 802.11n standard, which can achieve PDR greater than 90% while maintaining a *relaxed* congestion state as CBR remains below 30%. Table 5.8 shows results from using MCS with index 15 (`HtMcs15` in ns-3), corresponding to 64-QAM with 5/6 coding rate using 2 spatial streams. We notice that only when the probability of retransmission is 100% that CBR exceeds 30%. When we use MCS with index 31 (`HtMcs31`) which also uses 64-QAM modulation with 5/6 coding rate but with 4 spatial streams, we can achieve improved results as

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0684804	0.613823	0.00146857	1.86557	0.311501	0
0.10736	0.652204	0.00410321	0.817546	0.237057	0.1
0.163846	0.706674	0.00745557	0.385763	0.180189	0.25
0.181384	0.725172	0.00853044	0.359946	0.165869	0.3
0.250938	0.799295	0.0120064	0.208388	0.131877	0.5
0.326098	0.889195	0.0156269	0.15202	0.100189	0.75
0.338873	0.90574	0.016287	0.153022	0.0984112	0.8
0.362717	0.941613	0.0174906	0.143046	0.0944751	0.9
0.383347	0.976514	0.018585	0.132812	0.0833407	1

Table 5.7: Results from `OfdmRate54Mbps` Modulation. 4 Lanes, $\Delta x = 5, \Delta y = 3.5$

shown in Table 5.9, where the channel congestion state remains low, as CBR remains under 30% when repetition probability is 100%.

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0577907	0.647472	0.00117962	1.75588	0.269998	0
0.0874803	0.681583	0.0034762	0.639213	0.243251	0.1
0.130847	0.72993	0.00658779	0.353806	0.180362	0.25
0.144454	0.747372	0.00745927	0.307994	0.153865	0.3
0.198583	0.814851	0.0107854	0.210589	0.126028	0.5
0.258452	0.89814	0.0142565	0.150871	0.0939643	0.75
0.268913	0.914417	0.0148563	0.139356	0.093917	0.8
0.289155	0.947452	0.0159268	0.134782	0.0897637	0.9
0.306733	0.980504	0.0171331	0.123422	0.0846388	1

Table 5.8: Results from 802.11n modulations (High Throughput - `HtMcs15`). 4 Lanes, $\Delta x = 5, \Delta y = 3.5$

5.2.1.3 Results: Using 8 lanes, $\Delta x = 1, \Delta y = 1$ placement

This scenario is intended as a worst-case congestion scenario where all 500 vehicles are within range of one another. This eliminates the effect of the *hidden node* problem that would occur in the results in Section 5.2.1.2, where two nodes that are not within communication range of one another, would have a higher probability of simultaneous broadcast of wireless frames.

In this setup, when using `OfdmRate12MbpsBW10MHz` modulation in 10 MHz channel with no blind retransmissions, the two nodes of interests in the separated nodes scenario did not register any reception between the designated sender and receiver, which is the same as the previous setup. Increasing the blind retransmission rate to 0.5 improves the interarrival times, but we were not able

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0570782	0.639042	0.00121052	1.1937	0.318279	0
0.084862	0.672932	0.00361487	0.651964	0.229229	0.1
0.125564	0.723396	0.00677551	0.351725	0.17709	0.25
0.138342	0.741292	0.00769345	0.319523	0.171062	0.3
0.188949	0.807269	0.0110641	0.200024	0.126803	0.5
0.245007	0.89375	0.0145702	0.146793	0.104651	0.75
0.254849	0.910691	0.0152345	0.141666	0.101229	0.8
0.273529	0.943766	0.0162654	0.131413	0.0950335	0.9
0.290406	0.977222	0.017374	0.120257	0.0868555	1

Table 5.9: Results from 802.11n modulations (High Throughput - `HtMcs31`). 4 Lanes, $\Delta x = 5, \Delta y = 3.5$

to achieve a packet delivery ratio greater than 90% with all repetition probabilities, and the average CBR becomes high, as shown in Table 5.10. Using the highest modulation available for 802.11p, `OfdmRate27MbpsBW10MHz`, we can achieve a relatively higher PDR as shown in Table 5.11.

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.127387	0.386511	0.00361387	∞	0.480697	0
0.273112	0.441806	0.00954424	1.28824	0.369038	0.1
0.469546	0.524755	0.0160424	0.527514	0.23046	0.25
0.523664	0.551693	0.0178427	0.421341	0.242815	0.3
0.643988	0.65377	0.023337	0.374636	0.184224	0.5
0.67428	0.769703	0.0282668	0.646707	0.153699	0.75
0.676098	0.790117	0.0291192	0.70525	0.142548	0.8
0.678947	0.831409	0.0304055	0.844623	0.137848	0.9
0.680495	0.873329	0.0315755	0.791091	0.137511	1

Table 5.10: Results from `OfdmRate12MbpsBW10MHz` Modulation, 8 Lanes, $\Delta x = 1, \Delta y = 1$

Better results are achievable with the modulations/coding schemes that were introduced with the 802.11n standard (High Throughput). Using the middle modulation of index 15, `HtMcs15`, results in PDR greater than 90% when the probability for retransmission is 0.75, while maintaining average CBR lower than 30%, even when repeating every frame for additional times, as shown in Table 5.12, where it seems that the way we implement time separation of repeated frames impacts the results, as it reduces the average CBR.

Finally, using the highest modulation coding scheme available for 802.11n, `HtMcs31`, results are summarized in Table 5.13, and we notice that the average CBR does not reach 30%, even when every frame is repeated 5 times. The most significant improvement in performance is the reduction

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0840237	0.472771	0.00194093	0	0.383773	0
0.160892	0.521356	0.00636809	1.26672	0.284666	0.1
0.270054	0.593907	0.0117226	0.467828	0.200294	0.25
0.303039	0.617422	0.0131991	0.405787	0.189677	0.3
0.412782	0.712664	0.0181335	0.278599	0.140379	0.5
0.481784	0.823172	0.0226527	0.293641	0.114728	0.75
0.488906	0.845399	0.0233527	0.338076	0.114178	0.8
0.497937	0.887199	0.0246279	0.302661	0.109398	0.9
0.504712	0.926838	0.0259521	0.36153	0.101024	1

Table 5.11: Results from `OfdmRate27MbpsBW10MHz` Modulation, 8 Lanes, $\Delta x = 1, \Delta y = 1$

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0546246	0.703412	0.000952265	3.80731	0.27664	0
0.0845647	0.73391	0.00278238	0.953706	0.205154	0.1
0.128032	0.775359	0.00521431	0.401297	0.16676	0.25
0.141657	0.787028	0.00595035	0.375792	0.150004	0.3
0.196082	0.844982	0.00873945	0.223161	0.1209	0.5
0.257146	0.914935	0.0117269	0.167351	0.0933691	0.75
0.267724	0.928431	0.0123093	0.154479	0.0939084	0.8
0.288311	0.957426	0.0133066	0.133765	0.0866448	0.9
0.307086	0.984644	0.0143951	0.132626	0.0793685	1
					Additional Frames
0.263795	0.981396	0.00725469	0.236796	0.0697765	2
0.281422	0.987412	0.00661359	0.283541	0.0568958	3
0.300068	0.989983	0.005693	0.46706	0.0458718	5

Table 5.12: Results from `HtMcs15` Modulation, 8 Lanes, $\Delta x = 1, \Delta y = 1$

in the interarrival times between the two nodes of interest in the separate nodes scenario, where without any blind retransmissions, the average time between two receptions is in order of seconds, and it is reduced greatly, even when the probability of blind retransmission is low.

Both scenarios show similar results, and blind retransmission offers improvement in performance in terms of PDR. It does, however, introduce an increase in the average one-way delay as channel becomes busy with more transmission, but that negative effect is compensated by the improved interarrival times, and the observed one-way delay is slightly lower than that delay when using the 5G V2X technology.

AvgCBR	PDR	AvgOWD	Avg. interarrival Separate Nodes	Avg. interarrival Adjacent Nodes	Repeat Probability
0.0520806	0.708771	0.000905102	7.13723	0.282395	0
0.0801575	0.733743	0.0027144	0.930114	0.22155	0.1
0.12088	0.775713	0.00513017	0.409543	0.168362	0.25
0.133689	0.789683	0.00580172	0.375792	0.162702	0.3
0.185102	0.846745	0.00856803	0.221508	0.130781	0.5
0.242449	0.917132	0.0115728	0.155455	0.0955316	0.75
0.252678	0.930337	0.0121731	0.147307	0.0933289	0.8
0.272107	0.957973	0.0131373	0.140381	0.0900346	0.9
0.29007	0.98573	0.0142077	0.132834	0.0818564	1
					Additional Frames
0.25208	0.982407	0.007161	0.229834	0.0726262	2
0.269424	0.988701	0.00645283	0.307694	0.060826	3
0.288032	0.991302	0.00559934	0.446271	0.0474193	5

Table 5.13: Results from `HtMcs31` Modulation, 8 Lanes, $\Delta x = 1, \Delta y = 1$

5.2.2 Using DSRC+ in CACC

We used the same CACC controller described Section 3.1, where each experiment has 10 vehicles, and we attempt to find the minimum string stable time headway for every acceleration profiles. We used the `HtMcs31` modulation that is offered by 802.11n standard in 20 MHz channel. We ran two sets of tests with different blind retransmission probabilities of $P = 0.25$, and $P = 0.75$, and compare the results with the standard DSRC implementation using the default 10 MHz channel with the `OfdmRate6MbpsBW10MHz` modulation. We have already discussed the results from DSRC in Section 4.2 as well as the results from 5G NR V2X, and we summarize the results in Table 5.14.

Using DSRC+ with blind retransmissions, every sample tested can operate string stable platoons at lower time headway except for `sample1`. This in turn translates into higher traffic flow as vehicles could maintain smaller separation distance at lower time headway.

For the sinusoidal profile, the minimum string stable time headway is 0.8 seconds when using traditional DSRC, which achieves a traffic throughput of 2713 vehicles/hour. DSRC+ with retransmission probability of $P = 0.25$ can operate at a lower string stable time headway of 0.7, which translates into 2966 vehicle/hour, an improvement of 9.35%. Using a retransmission probability of $P = 0.75$ lowers the minimum string stable time headway to 0.6 seconds, achieving a traffic throughput of 3271 vehicle/hour, which is a 20.58% increase over traditional DSRC.

	Minimum string stable Headway			
Sample	**DSRC**	**DSRC+** $P = 0.25$	**DSRC+** $P = 0.75$	**5G NR V2X**
sin	0.8	0.7	0.6	0.7
step	1.1	0.8	0.7	1
us06	1.1	0.9	0.9	1
c2a	1.1	1.1	1	0.7
sample1	0.8	0.8	0.8	0.5
sample2	1	0.7	0.7	0.5
sample3	1.3	1.1	1	1.2
sample4	1.1	1	1	1
sample5	0.8	0.6	0.7	0.5

Table 5.14: Minimum string stable headway using DSRC+ in CACC experiments

The other synthetic profile that uses a step function, DSRC+ improvement has significantly higher improvement in traffic throughput than traditional DSRC. With DSRC+ using $P = 0.25$ and $P = 0.75$, the improvement is 18.29% and 33.17% over traditional DSRC respectively. The improvement in traffic throughput is summarized in Table 5.15. The average improvement in throughput is up to 15.76%.

	DSRC	**DSRC+** $P = 0.25$		**DSRC+** $P = 0.75$	
	Flow (vh/hr)	**Flow (vh/hr)**	**Improvement**	**Flow (vh/hr)**	**Improvement**
sin	2712.99	2966.68	9.35%	3271.38	20.58%
step	2270.34	2800.13	23.34%	3023.43	33.17%
us06	2450.27	2855.6	16.54%	2855.6	16.54%
sample1	3079.78	3079.78	0.00%	3079.78	0.00%
sample2	2418.36	3058.88	26.49%	3058.88	26.49%
sample3	2134.54	2418.98	13.33%	2584.93	21.10%
sample4	2418.98	2418.98	0.00%	2418.98	0.00%
sample5	2723.72	3217.16	18.12%	2947.78	8.23%

Table 5.15: Traffic flow improvement with DSRC+, compared to traditional DSRC

Chapter 6

Radio Access Technology Trade-offs

In this part, we try to investigate whether the choice of radio access technology (RAT) in VANET applications matter. There are two competing technologies: DSRC, which uses sensing-based CSMA channel access method, and Cellular-V2X, which uses technology a hybrid method that employs sensing of available resource in the wireless channel in addition to time-division.

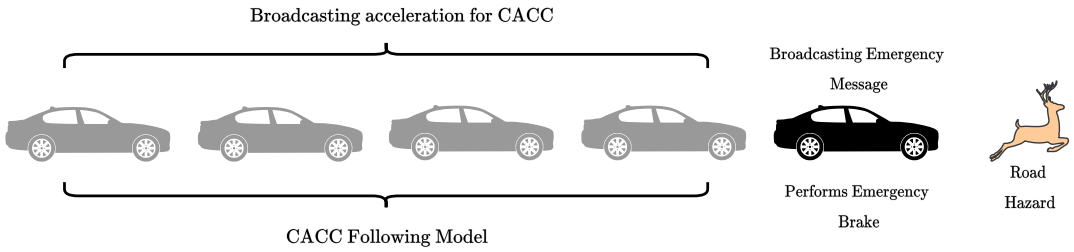
6.1 Experiment Setup

In our discussion in Section 4.3, 5G NR-V2X is a clear winner when it comes to packet delivery ratio (PDR) as it can achieve a PDR above 90% in a congested network of vehicle. Moreover, the use of 5G NR V2X in CACC application allows for the use of smaller separation margins as it exhibit better PDR than DSRC as shown in Section 4.2.

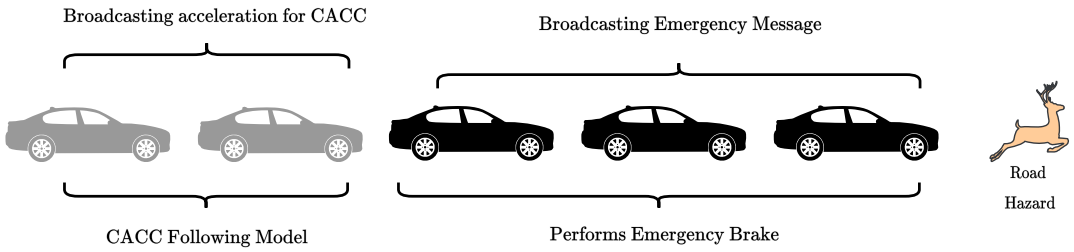
We focus our analysis on studying CACC platoons where emergency messages need to be disseminated to all platoon members. For this, we designed CACC simulations in ns-3, where the platoon leader accelerates to reach a maximum speed of 35 m/s, remains at that speed, and then after 60 seconds from the start of the simulation, it would detect road hazard, and enters an *emergency state*, where it performs an emergency brake, and starts broadcasting emergency messages every 100ms.

Recall that the CACC controller described in Chapter 3 would limit braking to -3.6576 m/s^2 (-12 ft/s^2), but we change that limit for vehicles in this emergency. Emergency braking can be expressed as a negative acceleration in terms of g, where $g = 9.8\text{m/s}^2$, and a realistic value for passenger vehicles is $0.8g$ as used in the literature [82].

Upon performing a sudden deceleration, the platoon leader also broadcasts an emergency message to other vehicles to alert them that it is making an emergency stop. It also stops broadcasting its acceleration that is used by the CACC controller because we are halting CACC operation and switching to emergency braking mode. When a vehicle receives an emergency message, it also enters an emergency state where it performs emergency braking, and in turn, broadcasts emergency messages instead of its acceleration. This is illustrated in Figure 6.1.



(a) Platoon leader detects road hazard



(b) Vehicles that received emergency message would rebroadcast the message

Figure 6.1: CACC setup for dissemination of emergency message

To evaluate our work, we use two metrics:

1. **Time-to-notify all:** the time it takes from the moment a platoon leader realizes an emergency situation to the moment the last vehicle in the platoon is notified of that emergency. In our simulation, an emergency situation is initiated at the 60th second of simulation time, where the platoon leader performs an emergency brake, and broadcasts an emergency message. Whenever a vehicle receives an emergency message, it records the time it received that

message, so the time to notify all platoon members of the emergency is the time recorded by the last vehicle to receive that message.

2. **Risk assessment:** for risk assessment, we would like a metric that quantifies exposure to the risk of collision. We use the Time-To-Collision TTC, which for a vehicle i and time instant k is defined as:

$$TTC_i(k) = \begin{cases} \dfrac{x_{i-1}(k) - x_i(k) - L}{v_i(k) - v_{i-1}(k)} & \text{if } v_i(k) > v_{i-1}(k) \\ \infty & \text{if } v_i(k) \leq v_{i-1}(k) \end{cases} \tag{6.1}$$

A TTC value of 3 seconds or less can be considered risky [54], so we can take that as a time-to-collision threshold denoted by TTC^*. TTC is evaluated for every vehicle in the platoon. For an assessment of an entire platoon of N vehicles, we use the Time-Exposed Time-to-collision, TET, which represents the time vehicles are exposed to risky TTC values that are under the threshold TTC^*. TET for a platoon of N vehicles at a time instant k is defined as:

$$TET(k) = \sum_{i=1}^{N} \delta_k \times \Delta k, \qquad \delta_k = \begin{cases} 1 & \text{if } 0 < TTC_i(k) \leq TTC^* \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

where δ is the switching variable, and Δk is the time scan interval. For a simulation period of T, we get

$$TET = \sum_{k=1}^{T} TET(k) \tag{6.3}$$

We want to compare simulations that use DSRC and 5G NR V2X under the emergency situation that we defined here. Since both use the same simulation time and time interval, we do not need Δk from Equation 6.2 for the sake of comparison.

Simulation is set up by varying number of vehicles to 5, 10, 15, 20, 25, 30, 35, 40, 45 or 50 so that we can study the behavior with platoons of different length. We use ns-3 nodes to represent vehicles, and in each vehicle, we installed a modified version of the CACC application that we described earlier in Section 3.3. The modified application performs CACC operations as before, where every vehicle broadcasts its acceleration every 100ms, and vehicles adjust their acceleration according to

the perceived acceleration of the preceding vehicle, i.e. using Equation 3.3. However, we schedule an event where the platoon leader enters into an *emergency state*, where it stops broadcasting its acceleration, and instead broadcast a special emergency message to platoon members every 100ms; the platoon leader also performs emergency braking at this point. Whenever a vehicle learns of that emergency, it enters into emergency state, broadcasts emergency messages, and initiates emergency braking. In real-world scenarios, emergency braking can range from $0.6g$ [44] to $1g$ [51], so we performed experiments with different deceleration values of $0.6g$, $0.8g$ and $1.0g$, where $g = 9.8m/s^2$.

To prevent the radio devices from starting at the exact same time in our simulation, ns-3 provides a way where applications installed in vehicles would start at different times with a function called `StartWithJitter`. An application installed in a vehicle would generate a random number between 0 and a maximum value that we call jitter max, or $J_{\max}$. We performed four sets of experiments with different values for $J_{\max}$ set to 25μs, 1000μs, 25000μs, and 100000μs.

The CACC experiments were also performed with varying values for time headway h set to 0.2s, 0.4s, 0.6s, 0.8s, 1.0s, 1.5s, and 1.8s. Therefore, the input parameters are summarized as follows:

1. n, number of vehicles in the CACC platoon.
2. Emergency braking value.
3. $J_{\max}$, the maximum jitter time used to randomize the starting times for applications installed in the vehicles.
4. h, time headway for the CACC controller in the vehicles.

For each set of input parameters, the experiment is repeated 50 times with the same parameters but with different seeds that ns-3 uses to generate random numbers. This allows us to minimize the variance in the randomness of our simulation runs. In total, we ran 84,000 simulations using Clemson's Palmetto cluster.

6.2 Results: Notification Time

6.2.1 Latency for platoons with smaller headway margins

Our ns-3 simulations shows that the two technologies appear to have similar latency performance with shorter platoons. The sets of experiments that use a starting jitter max $J_{\max}$ of 25μs represent the worst case scenario for the contention-based DSRC technology as the radios starting time are very close to one another. At lower time headway values h of 0.2 and 0.4 seconds, the platoon members maintain very low separation distance and that influences the time needed to propagate message as distance is smaller. Figures 6.2 and 6.3 show comparisons of the two technologies using $h = 0.2$ and $h = 0.4$ respectively, comparing the time needed to notify all vehicles of the platoon. The plot marker points represent the average latency value obtained from 50 simulation runs with different random-number generator seeds. We also shade the area between the lowest and highest latencies observed for those 50 run, where light red shading is for DSRC whereas light blue shading is for 5G NR V2X. We can observe that the two technologies have similar latency for shorter platoons of 5 and 10 vehicles. 5G NR V2X achieves slightly lower latency than DSRC for shorter platoon as well, but needed more time with longer platoons.

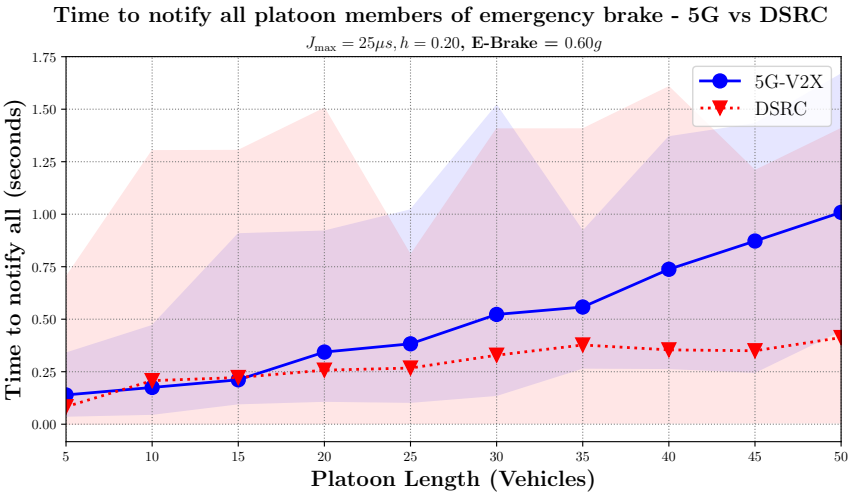


Figure 6.2: Time required to notify all platoon members - $h = 0.2$, e-brake $= 0.6g$, $J_{\max} = 25\mu$s

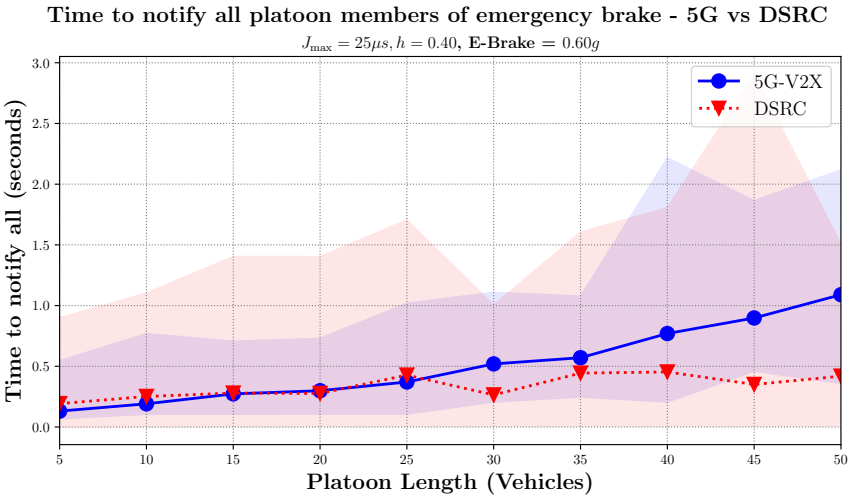


Figure 6.3: Time required to notify all platoon members - $h = 0.4$, e-brake $= 0.6g$, $J_{\max} = 25\mu$s

Due to the low $J_{\max}$ value of 25 μs, the DSRC experiment exhibit wider variance due the varying likelihood of packet drop due to collisions. To further minimize the effect of randomness of simulations, we removed experiments that had latency values that are considered as outliers. A sample is an outlier if it falls outside the Interquartile Range (IQR), that is values that are outside the range between the 25th and 75th percentiles. We plot the results from experiments that use small separation margins with time headway $h = 0.4$ again, and show the result in Figure 6.4.

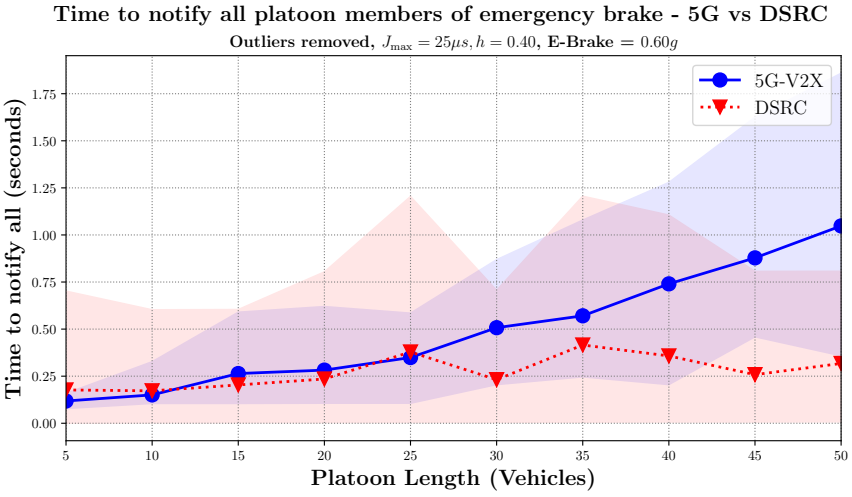


Figure 6.4: Time required to notify all platoon members - $h = 0.4$, e-brake $= 0.6g$, $J_{\max} = 25\mu$s, outliers outside IQR removed

In Figure 6.5, we present results where the random starting jitter $J_{\max}$ is set to 100,000 μs (or 100ms), and a smaller h value of 0.4 seconds. We have the same observation that DSR and 5G have comparable latency for shorter platoon, while 5G NR V2X requires significantly more time to notify all vehicles for longer platoons. For a platoon of 50 vehicles, 5G NR V2X needed an average of 1 seconds whereas DSRC needed a little under 250 ms. We also notice from the shaded area for DSRC that it achieves smaller variance, meaning that the variance of starting jitter times does impact the variance in observed latency for DSRC. When we remove outliers and focus on statistically significant results, as shown in Figure 6.6, we also observe that the latency increases significantly for 5G NR V2X for platoons of 20 vehicles or more.

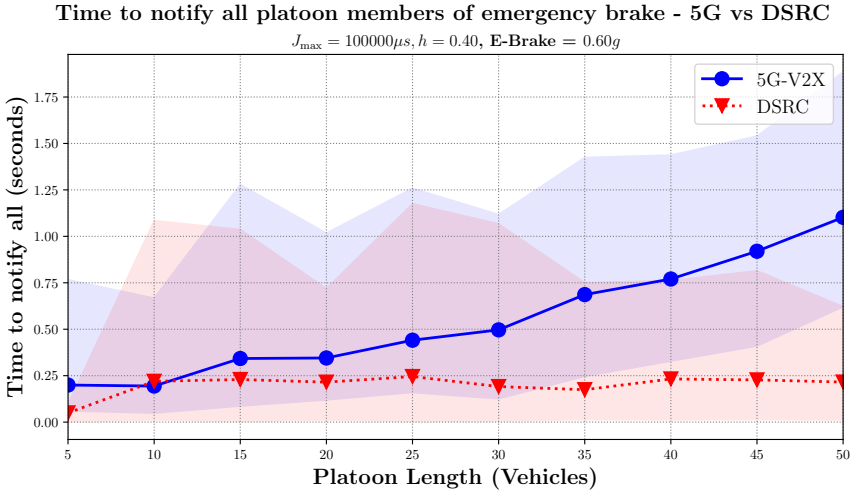


Figure 6.5: Time required to notify all platoon members - $h = 0.4$, e-brake $= 0.6g$, $J_{\max} = 100,000\mu$s

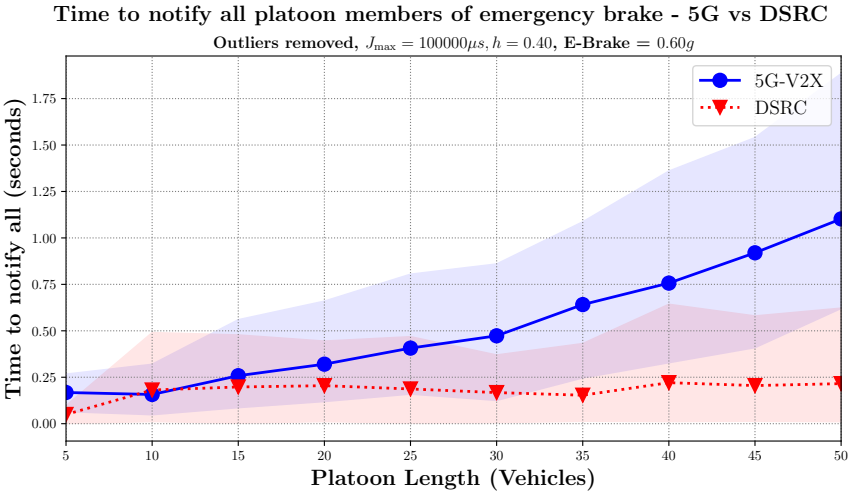


Figure 6.6: Time required to notify all platoon members of an emergency situation

Recall that we mentioned that we vary $J_{\max}$ to be 25μs, 1000μs, 25000μs, and 100000μs, running 50 experiment for each $J_{\max}$ value for a total of 200 experiments for a given h and emergency brake values.We combine the results from using all $J_{\max}$ values to create plots where each marker point represents the average of 200 experiments, and we show these results for $h = 0.4$ in Figure 6.7, and with outliers removed in Figure 6.8

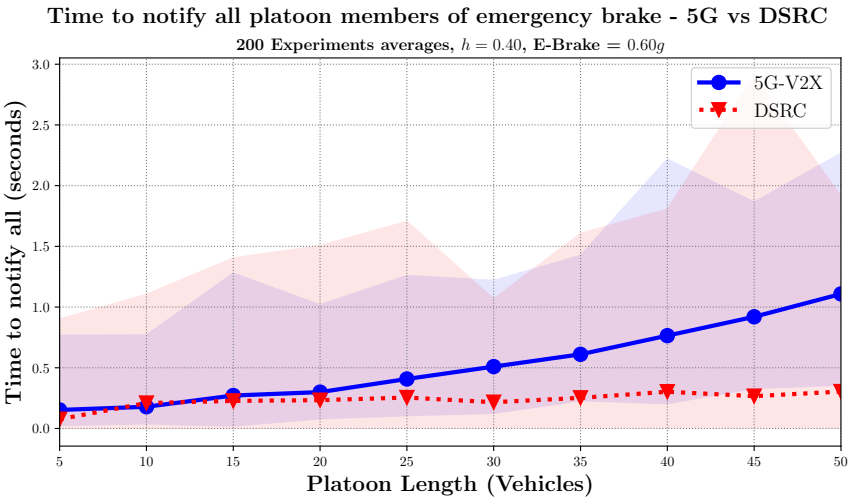


Figure 6.7: Time required to notify all platoon members - $h = 0.4$, e-brake $= 0.6g$, 200 Experiments per point

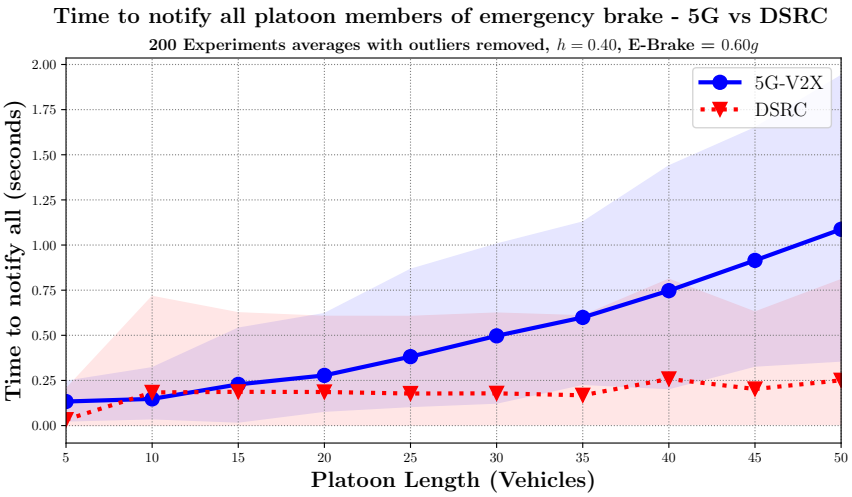


Figure 6.8: Time required to notify all platoon members - $h = 0.4$, e-brake $= 0.6g$, 200 Experiments per point outliers outside IQR removed

We draw the same conclusion from these results for CACC with smaller separation margin: For shorter platoons, the latency performance is similar between the two technology, where 5G NR V2X can achieve slightly lower latency for platoons shorter than 10 vehicles. DSRC achieves significantly lower latency for longer platoons of vehicles. We attribute the high latency in 5G NR V2X for longer platoons to the mechanism used for channel access, where vehicles that have wireless frames to broadcast would need to wait for a certain time slot to access the wireless channel, and the longer platoons increases the number of hops that emergency message go through until they cover the entire platoon. Therefore, we believe that this applies, not only to 5G NR V2X, but also any possible future technologies that employ channel access with time-division in which every device waits for its turn to use the wireless channel would incur additive increase of latency when the data frame needs to go over multiple hops. This latency is proportional to the number of hops from the originator of the message.

The results from the remaining experiments are presented in Appendix E, which show similar findings when using $h = 0.6$ and $h = 0.8$ in Figures 28 and 29 respectively. In addition, the intensity of emergency braking does not have any significant impact on the time needed to notify all vehicle.

6.2.2 Latency for platoons with larger separation margin

For CACC platoons that use larger time headway values ($h \geq 1.0$), vehicles maintain larger separation margins, which influences the successful transmission of wireless frames due to propagation loss over distance. For $h = 1.0$ and $J_{\max} = 25\mu$s, the performance is comparable between the two technologies, and DSRC achieves slightly lower latency than 5G NR V2X for the CACC platoons with 45 and 50 vehicles. The DSRC performance is better when removing outliers, indicating a higher level of variance for DSRC. This can be attributed to the use of very small starting jitter of only 25μs. These results are illustrated by Figure 6.9 showing a plot with all samples, and Figure 6.10 that plots the samples after removing outliers.

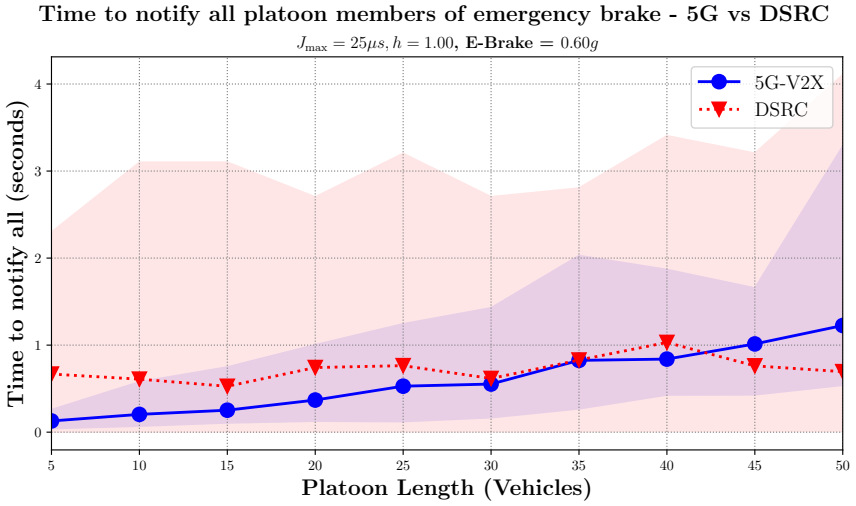


Figure 6.9: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, $J_{\max} = 25\mu$s

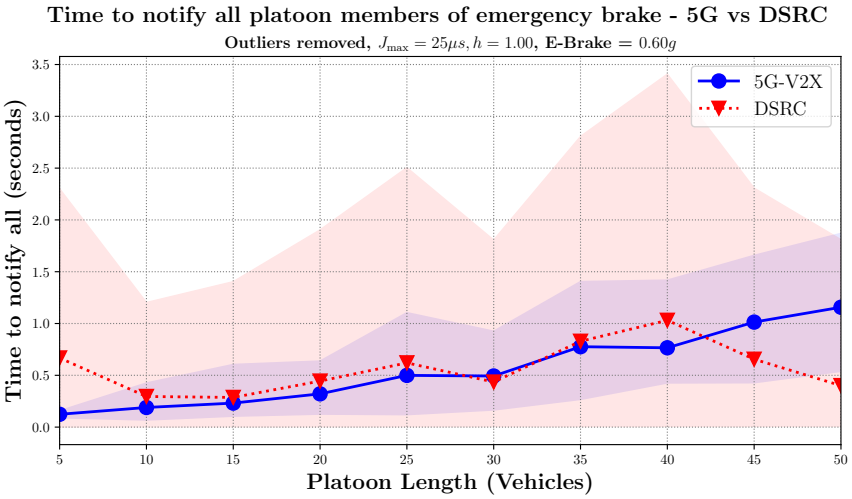


Figure 6.10: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, $J_{\max} = 25\mu$s, outliers outside IQR removed

Using a higher $J_{\max}$ of 100000μs (100 ms), DSRC performs better for all platoon lengths that were used, and the latency variance for 5G increases significantly, as shown Figure 6.11. Figure 6.12 presents a plot of the same data after removing outliers, indicating increased variance in the performance of DSRC.

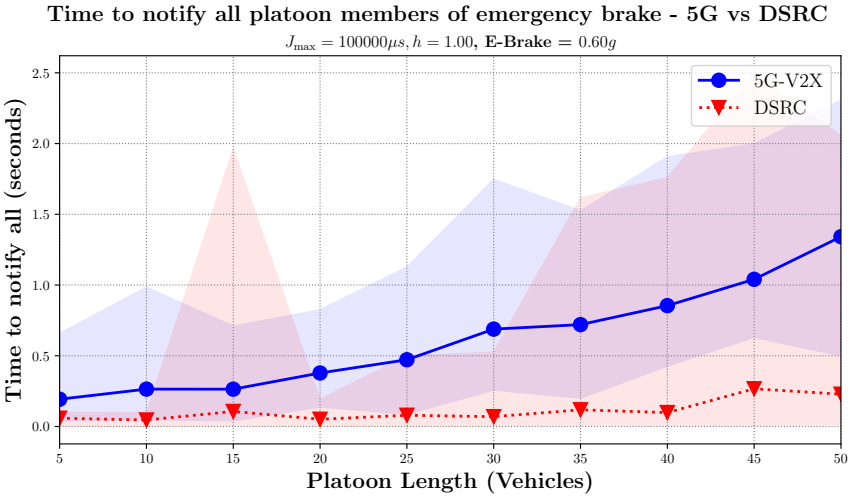


Figure 6.11: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, $J_{\max} = 100,000\mu$s

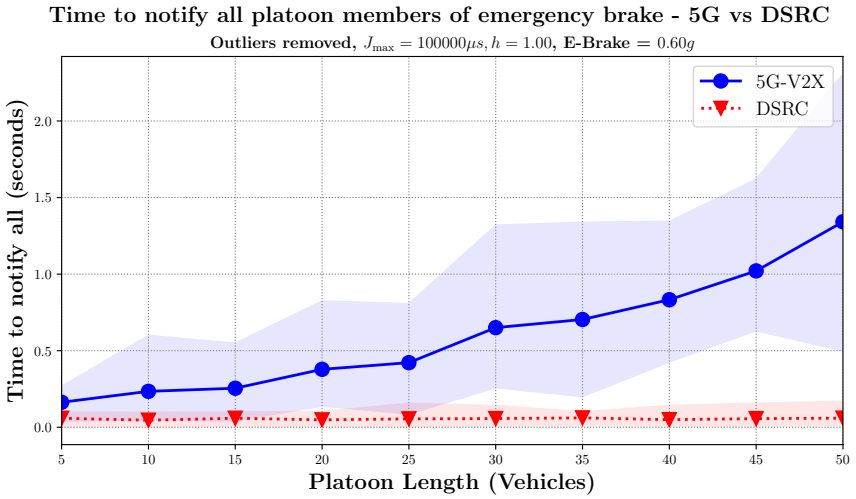


Figure 6.12: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, $J_{\max} = 100,000\mu$s, outliers outside IQR removed

Continuing with the same time headway of $h = 1.0$, we combine all four sets of experiments that have different $J_{\max}$, so every data point plotted is the average of 200 simulation samples, and show the results in Figure 6.13, and with outliers removed in Figure 6.14

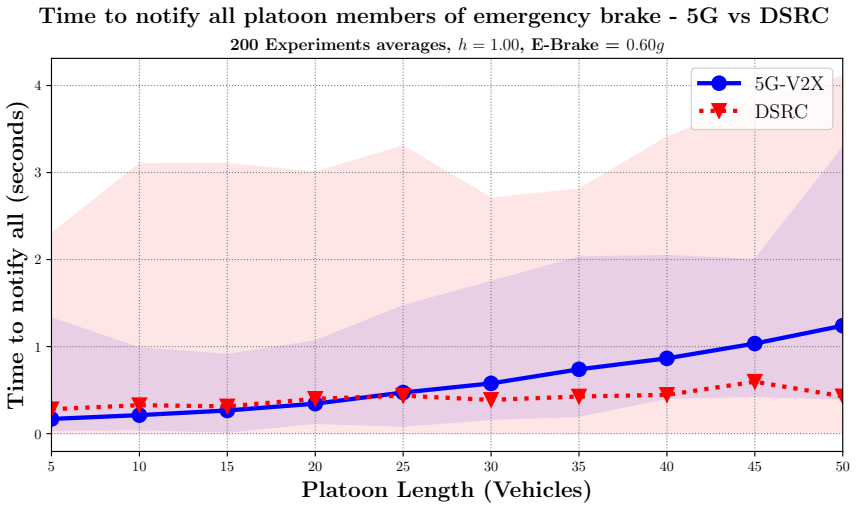


Figure 6.13: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, 200 Experiments per point

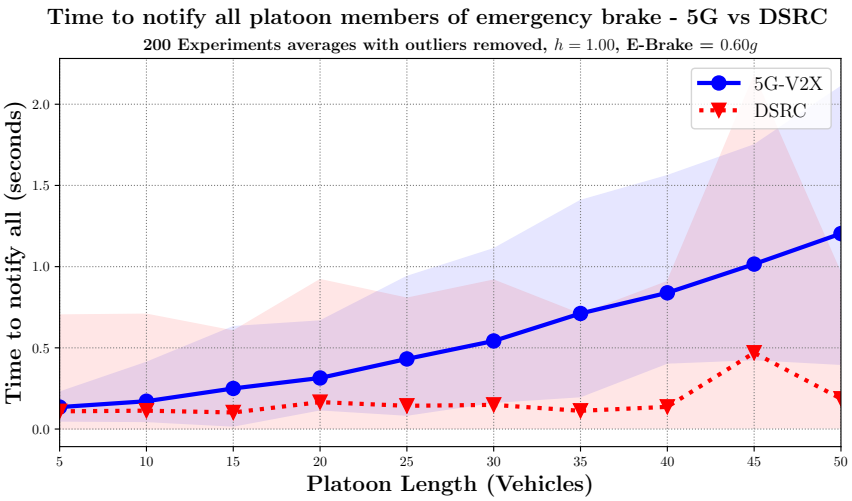


Figure 6.14: Time required to notify all platoon members - $h = 1.0$, e-brake $= 0.6g$, 200 Experiments per point with outliers outside IQR removed

We conclude that the added distance between platoon members results in a higher number of hops that significantly impacts the performance of 5G NR V2X. It also reduces the probability of collisions since vehicular density is lower, giving DSRC the advantage as it increases the probability of successful transmission, and DSRC vehicles do not need to wait as long as their 5G NR V2X counterpart.

6.3 Results: Collision Risk Assessment using TET

Another way to look at the impact of wireless technology is to study its impact on collision risk in CACC. The Time-Exposed Time-to-collision (TET) defined in Equations 6.3 serves as a good assessment of an entire simulation run. We choose Time-To-Collision threshold (TTC^*) to be 3 seconds as that is considered risky. All experiments were run with $\Delta k = 50$ms, meaning that we assess collision risk every 50 milliseconds. For every experiment, we recorded vehicles' positions and velocities every 50ms in order to compute TTC values using Equation 6.1 for every vehicle in every time step. Then we use Equation 6.2 to compute $TET(k)$ for all platoon members (the platoon leader will always have TTC values of ∞), and then for every time step k , we use Equation 6.3 to calculate a TET value for an entire simulation run.

6.3.1 TET for platoons with smaller headway margins

Starting applications with very low $J_{\max} = 25\mu$s exposes DSRC vehicles to higher risk of collision than 5G NR V2X, when we set emergency deceleration to $0.6g$, as shown in Figure 6.15. Increasing emergency deceleration to $0.8g$ and $1.0g$ also shows increased risk in DSRC experiments as vehicles would perform harsher braking, exposing vehicle to increased collision risk.

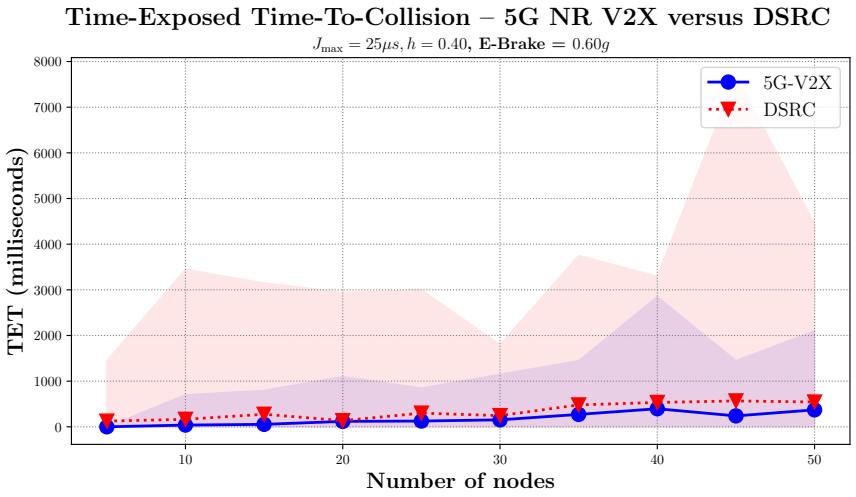


Figure 6.15: TET, $h = 0.4$, e-brake $= 0.6g$

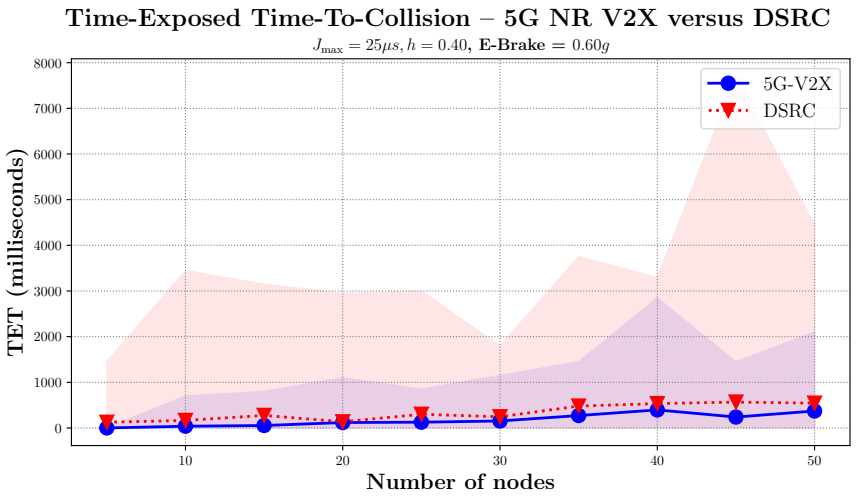


Figure 6.16: TET, $h = 0.4$, e-brake $= 0.6g$

However, this poor performance for DSRC devices is due to the use of very small time separations between the starting times of the radio devices, causing more packet losses, and therefore, exposing

vehicles to increased collision risk. The results illustrated by Figure 6.17 show that increasing $J_{\max}$ to 25,000 μs gives DSRC slightly better performance than 5G NR V2X, and variance in TET is comparable. We then remove statistical outliers from the results and plot the results shown in Figure 6.18, which shows that DSRC clearly performs better than 5G NR V2X as platoon length increases, with TET over 200ms for 5G NR V2X.

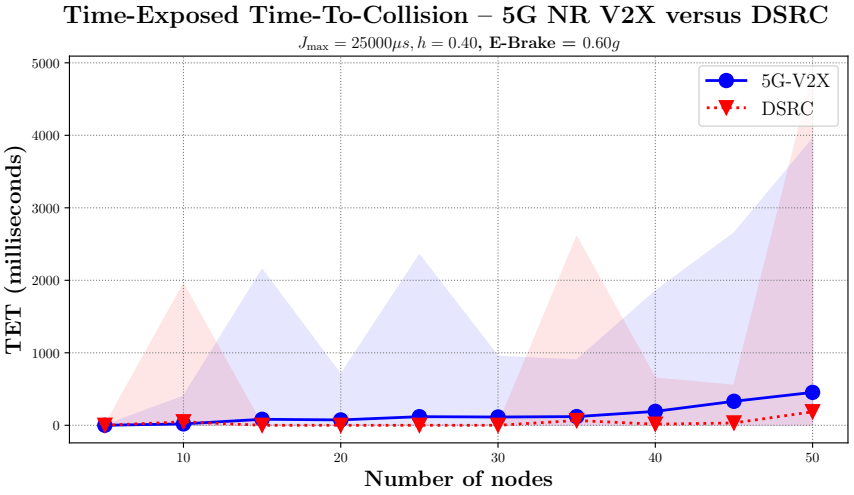


Figure 6.17: TET, $h = 0.4$, e-brake $= 0.6g$

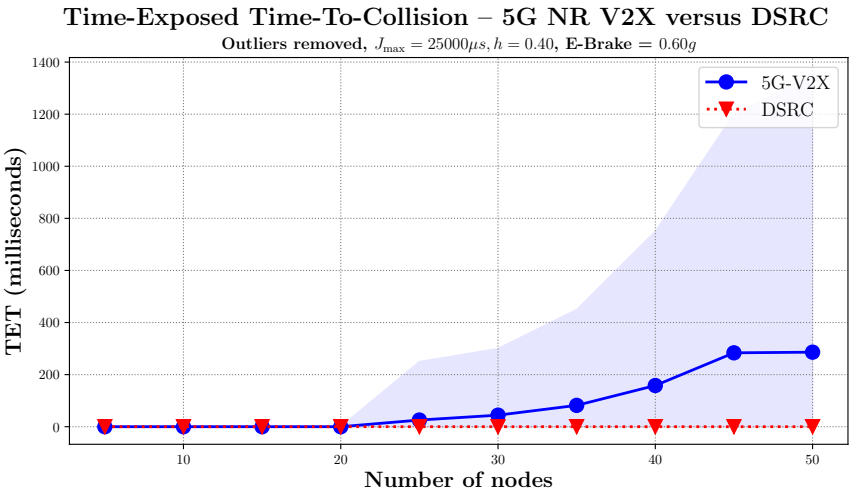


Figure 6.18: TET, $h = 0.4$, e-brake $= 0.6g$

When we combine results from 200 different simulation runs that use all tested $J_{\max}$, we get the results in Figure 6.21, and after removing outliers in Figure 6.20. We reach the same conclusion

in both cases: The two technologies have comparable exposure to collision risk for shorter platoons (under 20 vehicles), but the exposure to collision risk for 5G NR V2X increases significantly for longer platoons.

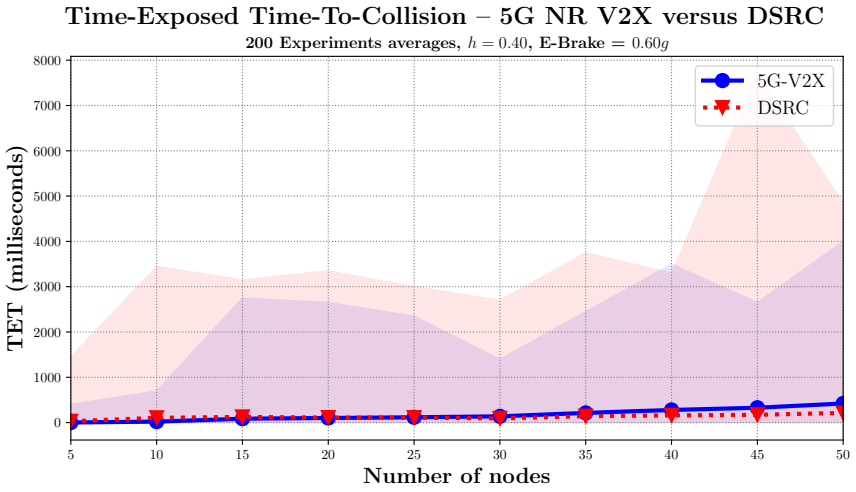


Figure 6.19: TET, $h = 0.4$, e-brake $= 0.6g$, 200 Experiments per point

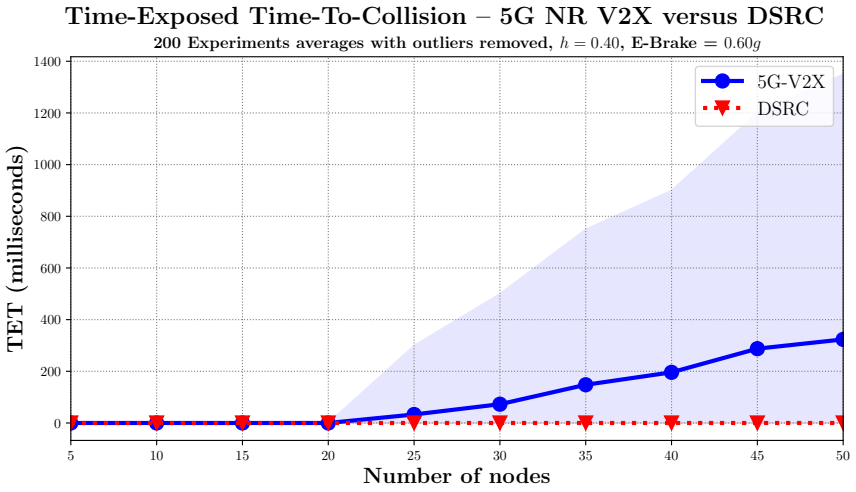


Figure 6.20: TET, $h = 0.6$, e-brake $= 0.6g$, 200 Experiments per point with outliers removed

We can also reach the same conclusion when using harsher emergency deceleration as shown in Figure 6.21 and Figure 6.22. Using even smaller time headway of $h = 0.2$, we reach similar conclusions as shown in Figure 33, and Figure 39 in Appendix F.

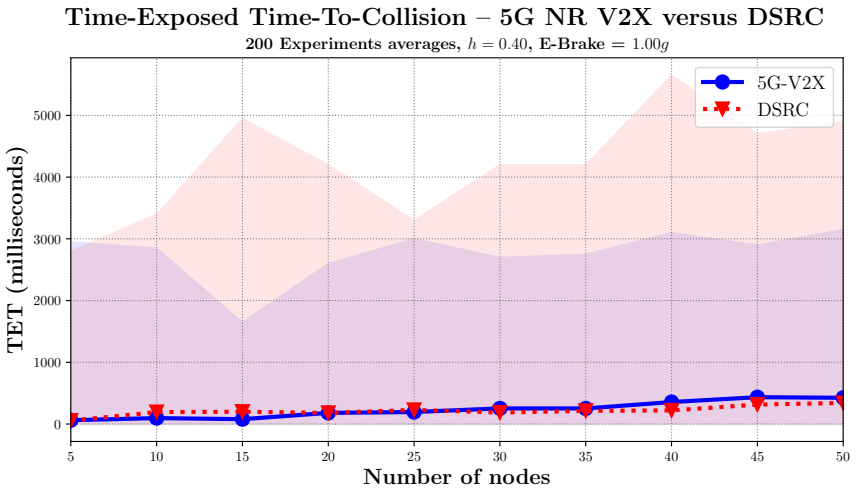


Figure 6.21: TET, $h = 0.4$, e-brake $= 1.0g$, 200 Experiments per point

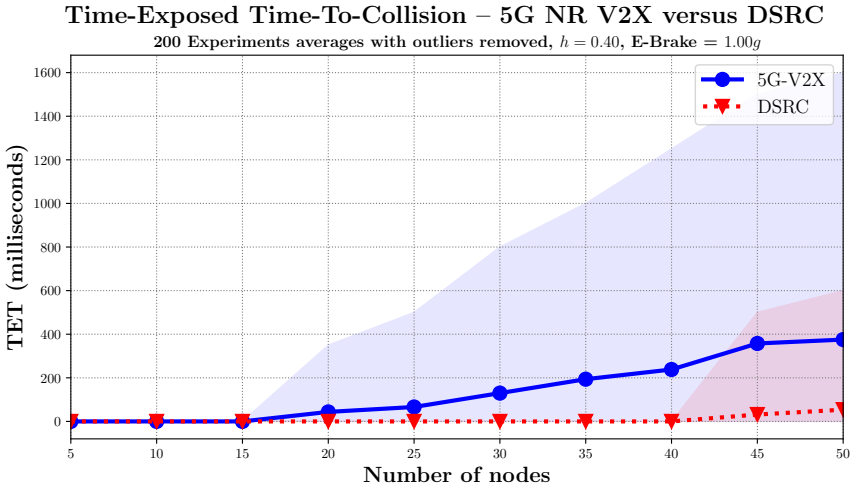


Figure 6.22: TET, $h = 0.6$, e-brake $= 1.0g$, 200 Experiments per point with outliers removed

6.3.2 TET for platoons with larger separation margin

Let us discuss what happens for CACC platoons that use larger time headway ≥ 1.0. Let us start with experiments where there is a smaller time variance in the starting time, with $J_{\max} = 25\mu$s. In this setup, 5G NR V2X achieves better performance than DSRC for experiments with all tested platoon length as shown in Figure 6.23 and 6.24. We can attribute this to the increase in separation distance and the potential consecutive packet loss that may happen as a result of starting the radio devices with small time variance. Therefore, the reliability of 5G NR V2X, that is, having higher

packet delivery ratio, helps it achieve safer operation with lower TET values than DSRC.

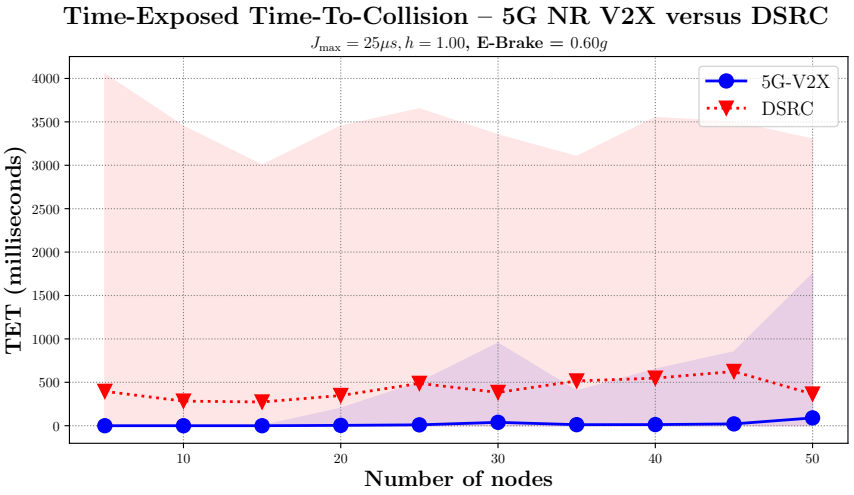


Figure 6.23: TET, $h = 1.0$, e-brake $= 0.6g$

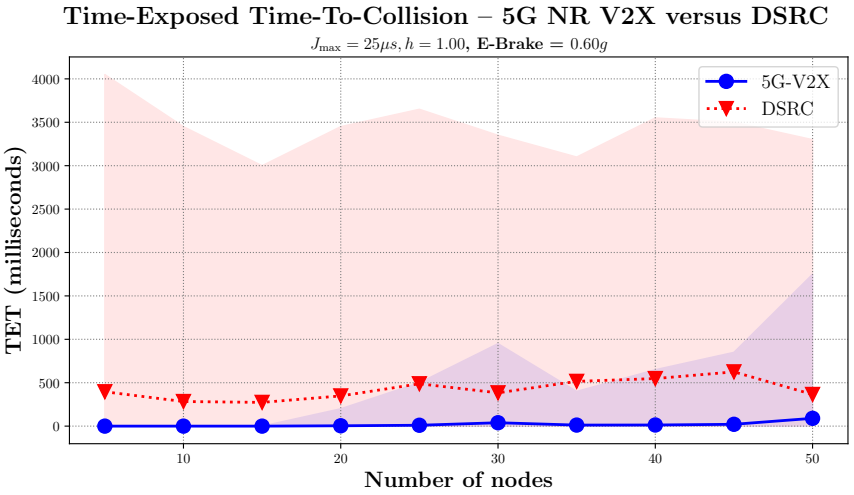


Figure 6.24: TET, $h = 1.0$, e-brake $= 0.6g$

For experiments with higher starting jitters $J_{\max}$ to 25000μs and $J_{\max}$ to 100000μs, we observe improved DSRC performance than the previous case with $J_{\max} = 25\mu$s. This is due to the fact that the probability of packet drops is lower because radio devices are started with higher time variance. We also observed similar results when we plot the combined experiments with all tested $J_{\max}$ values as shown in Figure 6.28.

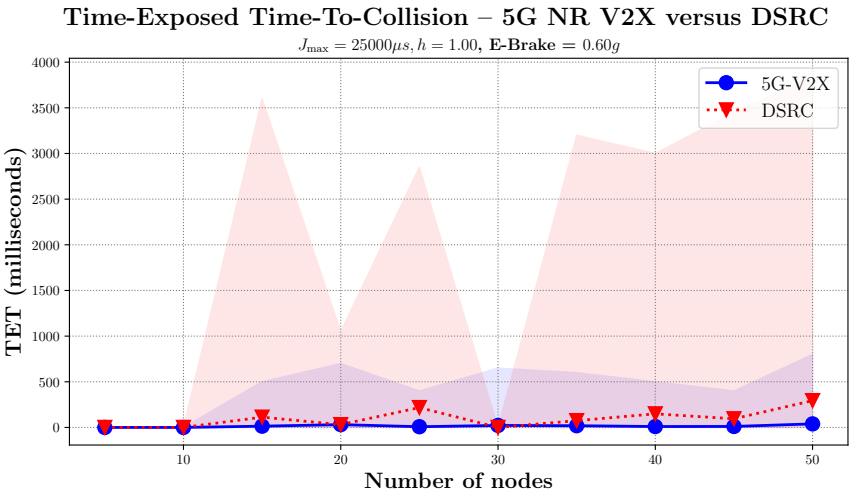


Figure 6.25: TET, $h = 1.0$, e-brake $= 0.6g$

When we remove outliers from the results with $h = 1.0$, the two technologies have the same level of exposure to collisions when using $J_{\max} = 25\mu$s for platoons with 30 vehicles or fewer, and DSRC becomes slightly more risky with longer platoons, as shown in Figure 6.26. The results with $J_{\max} = 25000\mu$s show that both technologies perform the same, with no exposure to risky condition as shown in Figure 6.26.

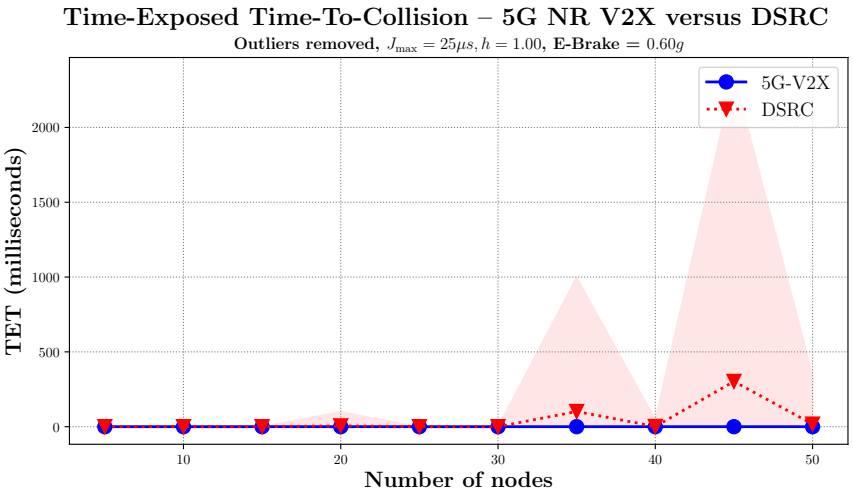


Figure 6.26: TET, $h = 1.0$, e-brake $= 0.6g$, outliers removed

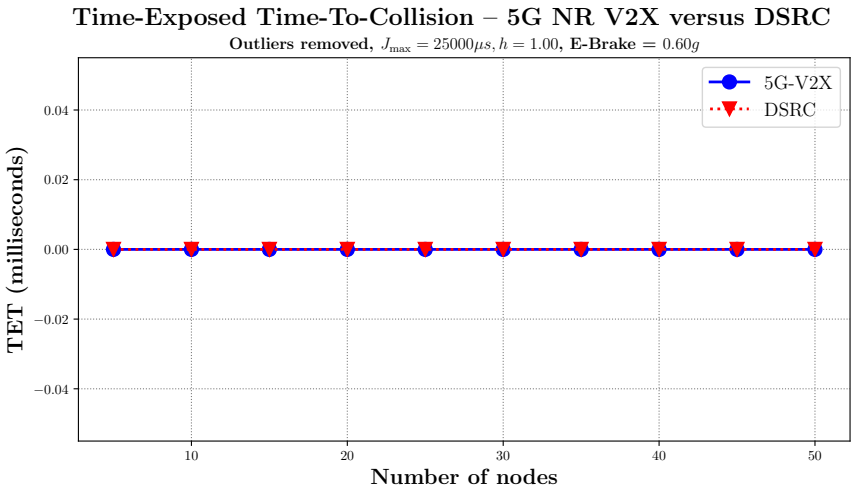


Figure 6.27: TET, $h = 1.0$, e-brake $= 0.6g$, outliers removed

When we combine all experiments from all $J_{\max}$ values, Figure 6.28 shows that DSRC have slightly worse performance than 5G NR V2X for all platoon length. However, when outliers are removed, the two technologies do not expose platoon vehicles to any risk as both achieve zero TET values for all the experiments as shown in Figure 6.29, even when using a very harsh emergency deceleration of $1.0g$, as shown in Figure 6.30

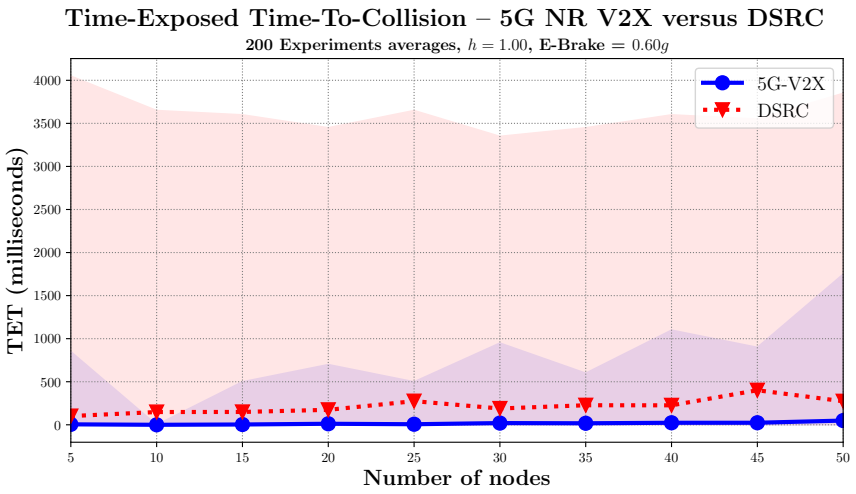


Figure 6.28: TET, $h = 1.0$, e-brake $= 0.6g$, 200 Experiments per point

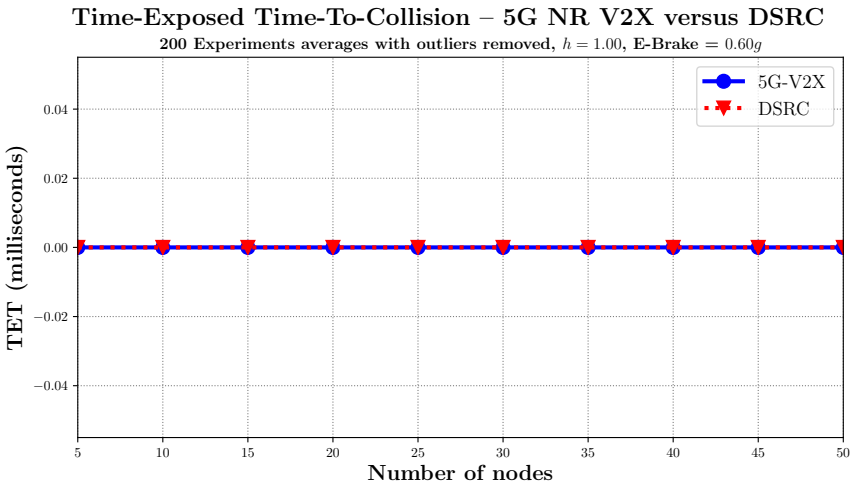


Figure 6.29: TET, $h = 1.0$, e-brake $= 0.6g$, 200 Experiments per point with outliers removed

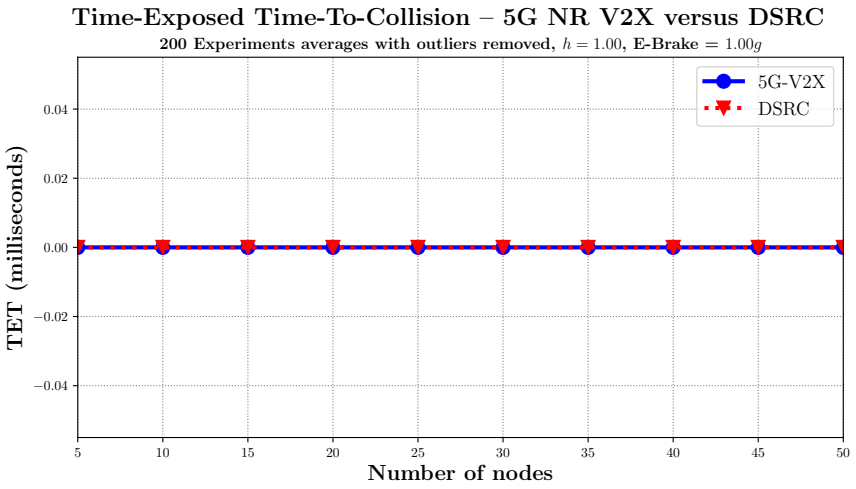


Figure 6.30: TET, $h = 1.0$, e-brake $= 1.0g$, 200 Experiments per point with outliers removed

When using even larger time headway ($h = 1.5$ and $h = 1.8$), the results are similar and shown in Appendix F in Figure 37, 43, 38, 44, and 44. We conclude that for platoons with relatively larger separation margins ($h \geq 1$), the risk of exposure to collisions is similar for statistically significant experiments.

Chapter 7

Conclusion

The introduction of DSRC in 1999 opened up the opportunity for real-world Vehicular Ad hoc Network (VANET) applications. However, the technology did not achieve as much adoption as hoped, and is now competing with another Radio Access Technology (RAT) that is based on cellular communication in the sidelink channel, and can be used for vehicle-to-vehicle communication.

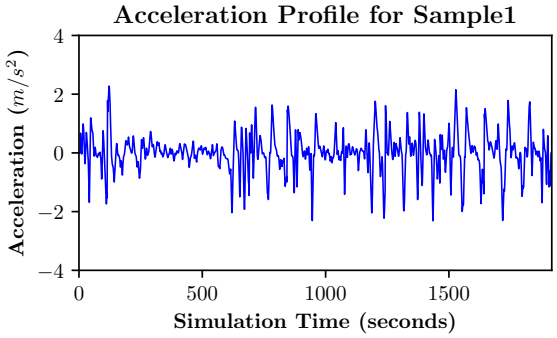
In this work, we studied the performance of VANET applications, with a focus on Cooperative Adaptive Cruise Control. We analyzed the performance of CACC application that use a one-vehicle look-ahead controller, using string stability as an evaluation metric. We established minimum string stable time headway for simulations scenarios under perfect network condition, then successfully implemented a dynamic time headway adaptation algorithm2e that adapts to network loss processes that are common in VANET. The adaptation algorithm improves vehicular traffic flow significantly against experiments that use the minimum string stable time headway for the same loss process, while also maintaining string stability throughout simulation, as discussed in Chapter 4.

We also proposed DSRC+ in Chapter 5, where we implemented simulation scenarios that utilize modern modulation/coding schemes that would be introduced in the IEEE 802.11bd amendment, and propose the use of blind retransmission to improve the packet delivery ratio. Our results show that DSRC+ can improve the packet delivery ratio of VANET that periodically broadcast packets, and also shows that it can improve the performance of CACC's traffic flow as it allows them to operate at lower time headway values.

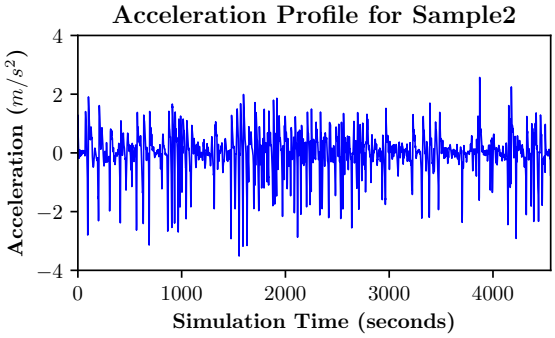
Finally, we studied the trade-offs in the choice between two RATs available today for VANET application in Chapter 6, where we operate CACC platoons where the platoon leader performs emergency braking, necessitating the dissemination of emergency messages to all platoon members. We used the time-to-notify all vehicles of this emergency, and a collision-risk assessment metric called Time-Exposed Time-to-collision (TET), where we consider vehicles with a Time-To-Collision (TTC) value of 3 seconds or under to be under a risky condition. Our findings make us reach the conclusion that 5G NR V2X, a technology with time-division channel access element in it, performs poorly relative to DSRC, when needing to disseminate a message through a long platoon. We also found that 5G NR V2X has noticeably increased risk of collision for longer platoons when operating at lower time headway values, whereas both technologies can operate with little to no collision risk when operating with larger separation margins when using higher time headway values.

Appendices

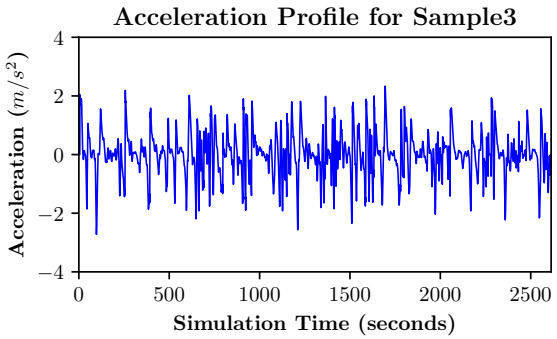
Appendix A Real vehicle acceleration profiles



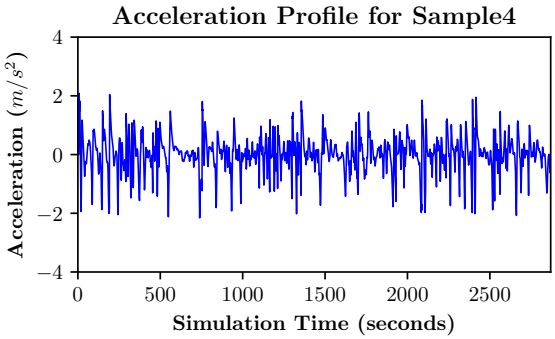
(a) Sample 1 Acceleration



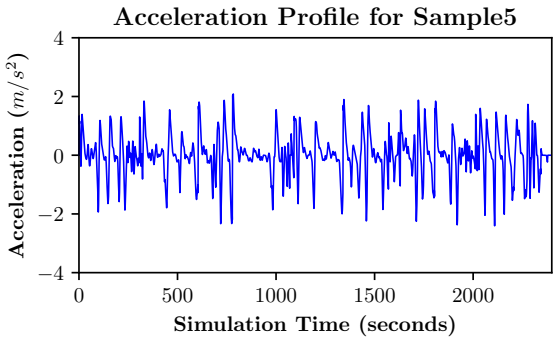
(b) Sample 2 Acceleration



(c) Sample 3 Acceleration



(d) Sample 4 Acceleration



(e) Sample 5 Acceleration

Figure 1: Acceleration Profiles

Appendix B Effect of headway time on traffic throughput

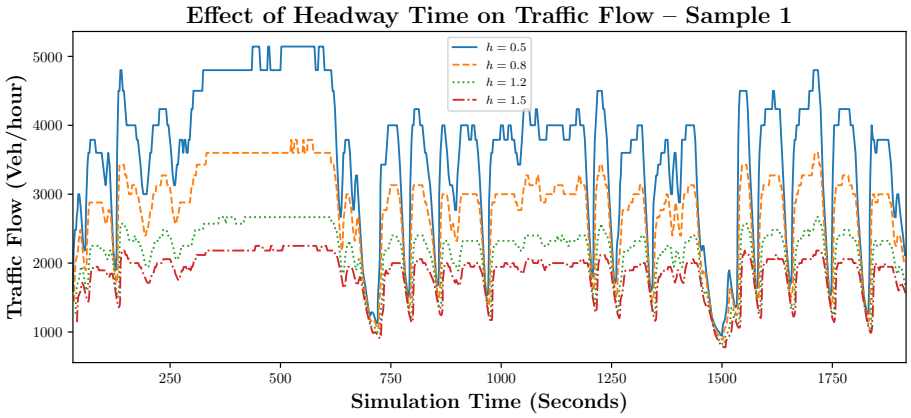


Figure 2: Traffic Throughput for Sample 1 with different headway values

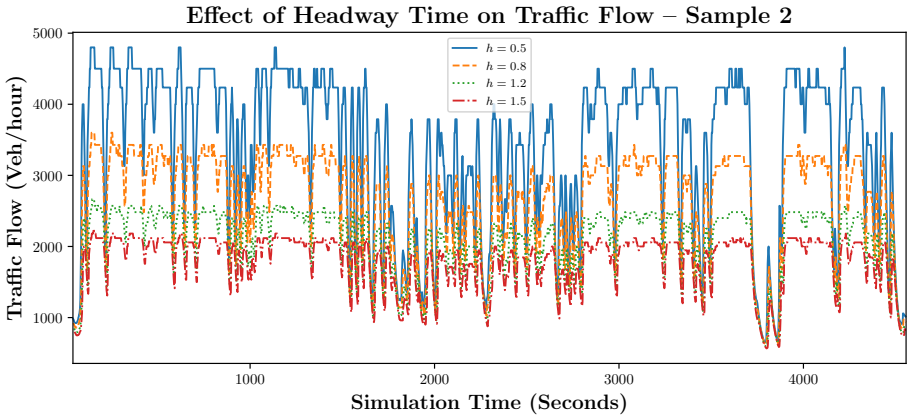


Figure 3: Traffic Throughput for Sample 1 with different headway values

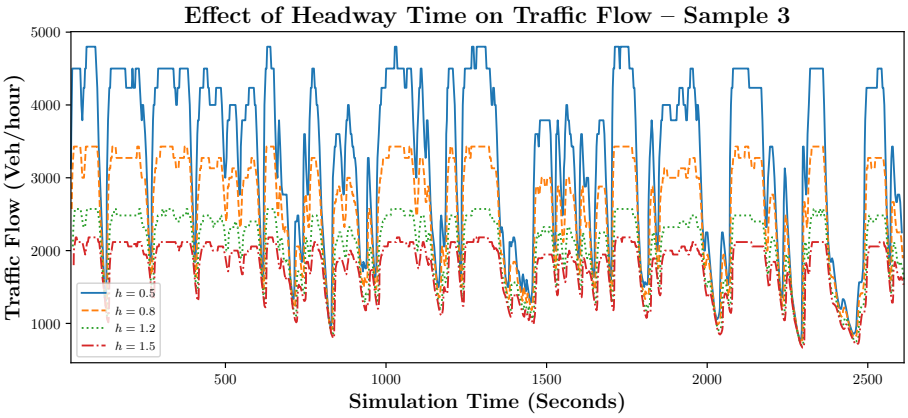


Figure 4: Traffic Throughput for Sample 3 with different headway values

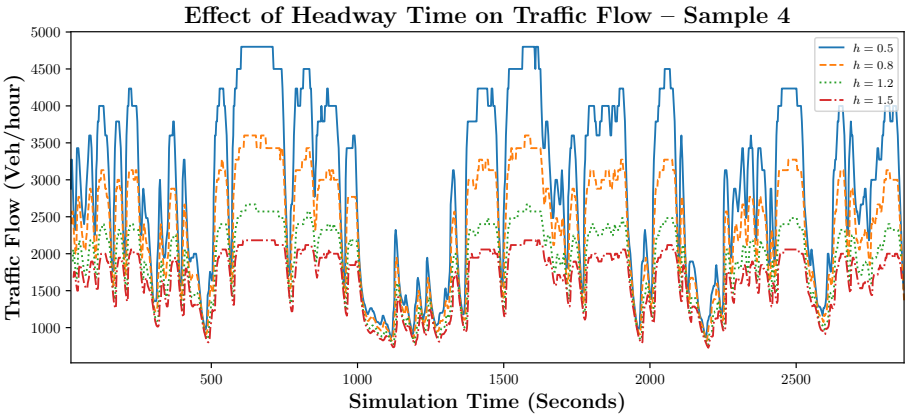


Figure 5: Traffic Throughput for Sample 4 with different headway values

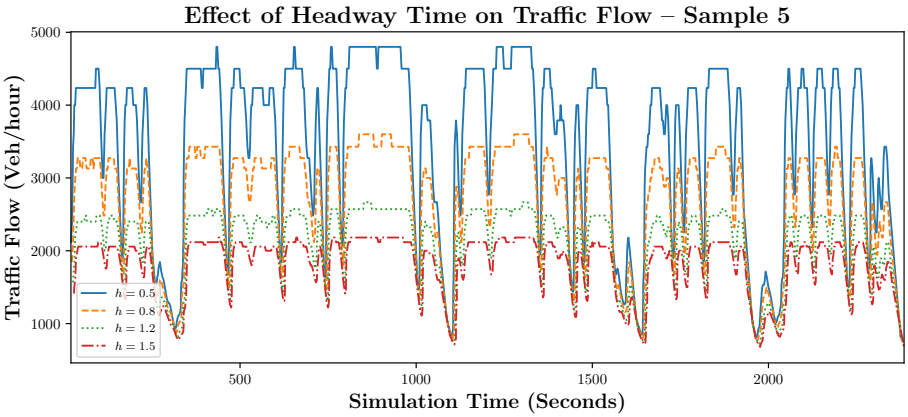


Figure 6: Traffic Throughput for Sample 5 with different headway values

Appendix C Improved acceleration estimation using Kalman filter

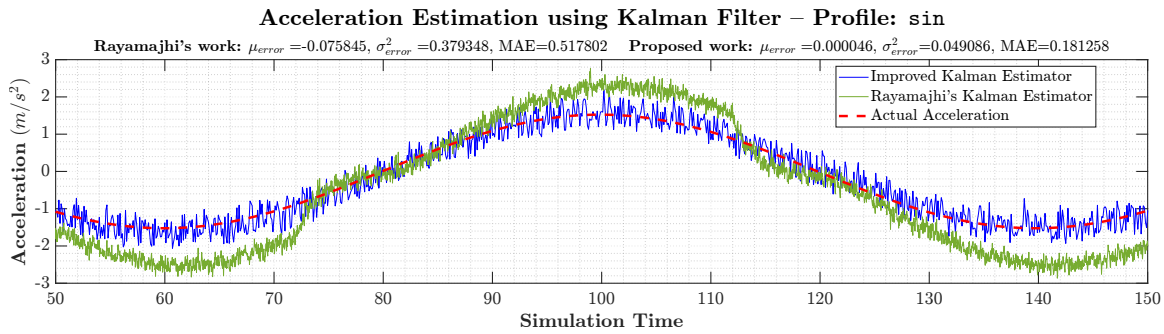


Figure 7: Improvement in acceleration estimation using sinusoidal acceleration profile

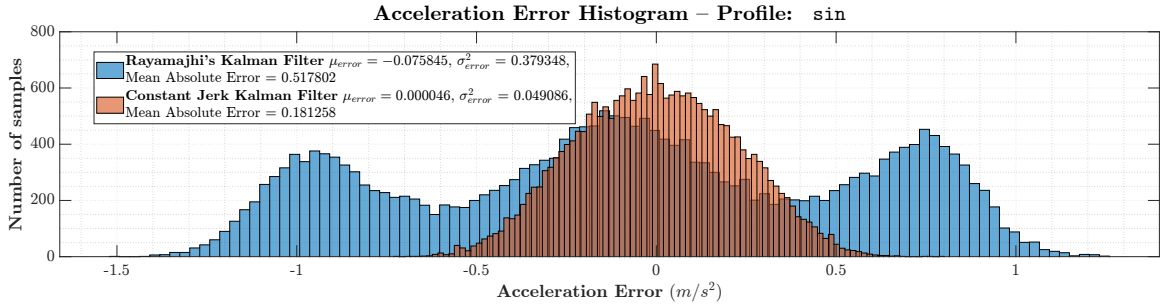


Figure 8: Histogram of acceleration estimation errors for sinusoidal acceleration profile

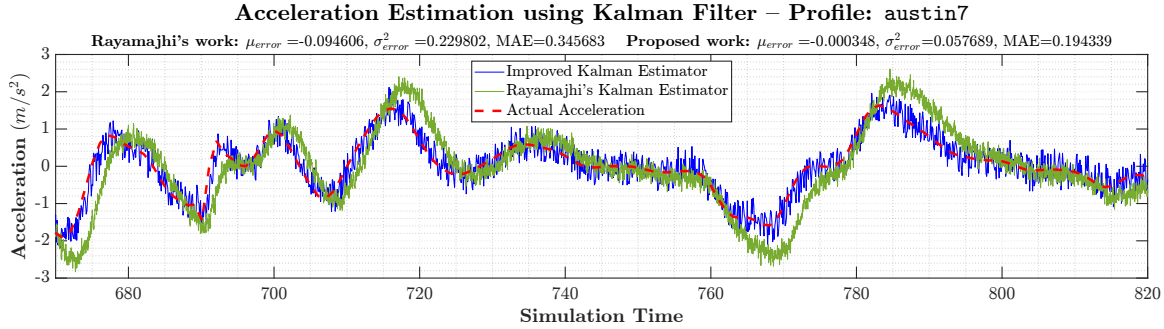


Figure 9: Improvement in acceleration estimation using `sample1` (`austin7`) acceleration profile

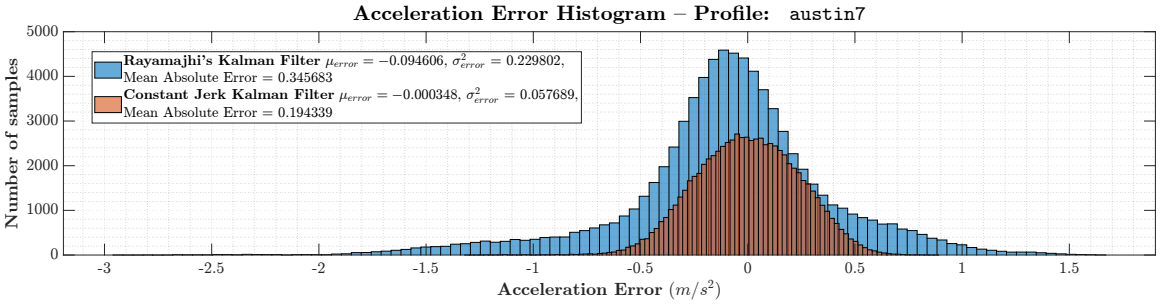


Figure 10: Histogram of acceleration estimation errors for `sample1` (`austin7`) acceleration profile

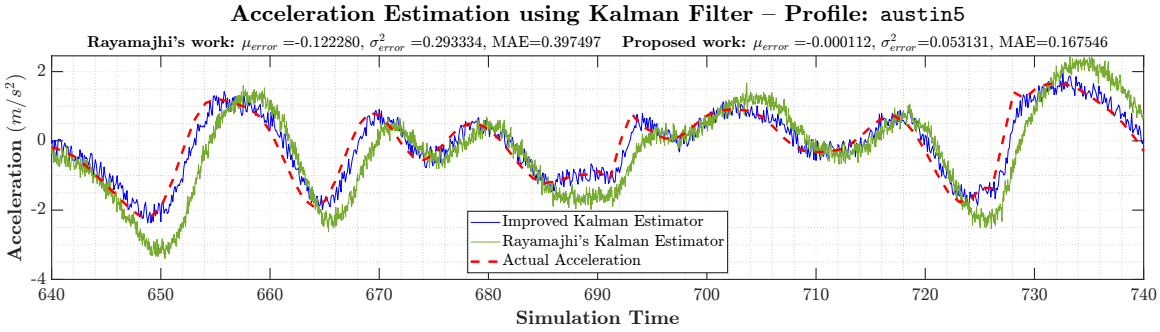


Figure 11: Improvement in acceleration estimation using `sample3` (`austin5`) acceleration profile

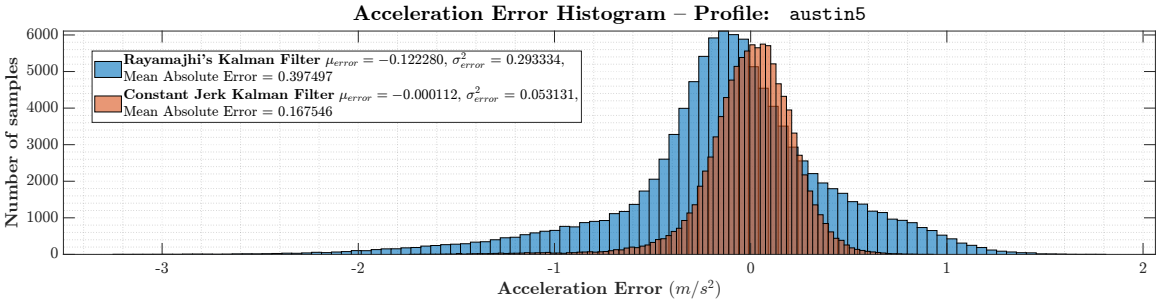


Figure 12: Histogram of acceleration estimation errors for `sample3` (`austin5`) acceleration profile

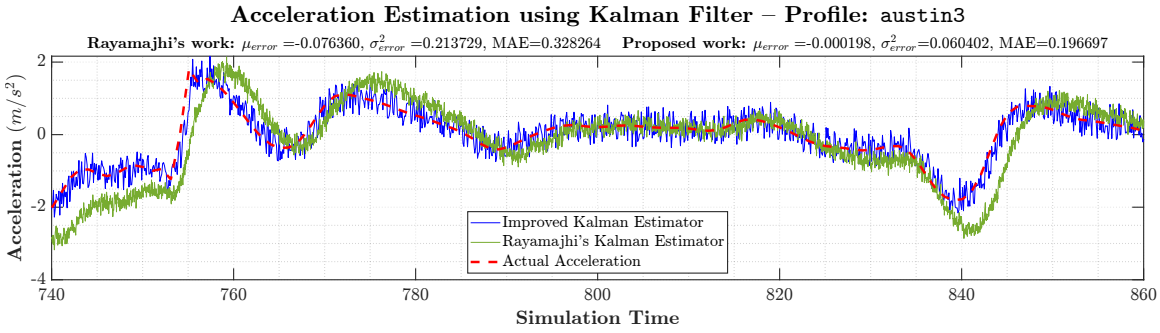


Figure 13: Improvement in acceleration estimation using `sample4` (`austin3`) acceleration profile

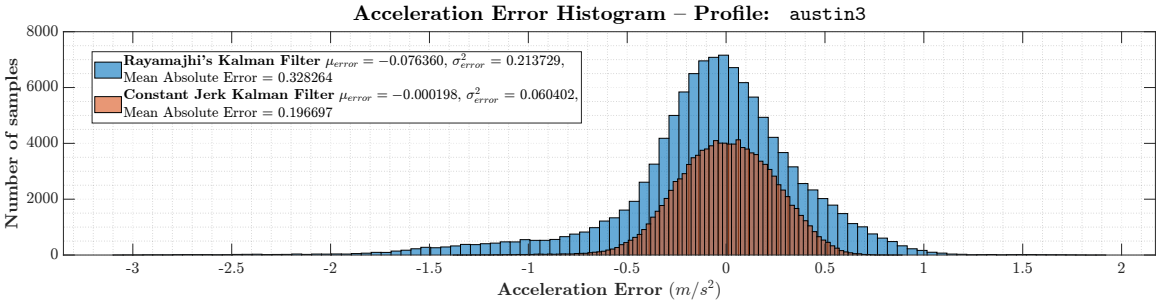


Figure 14: Histogram of acceleration estimation errors for `sample4` (`austin3`) acceleration profile

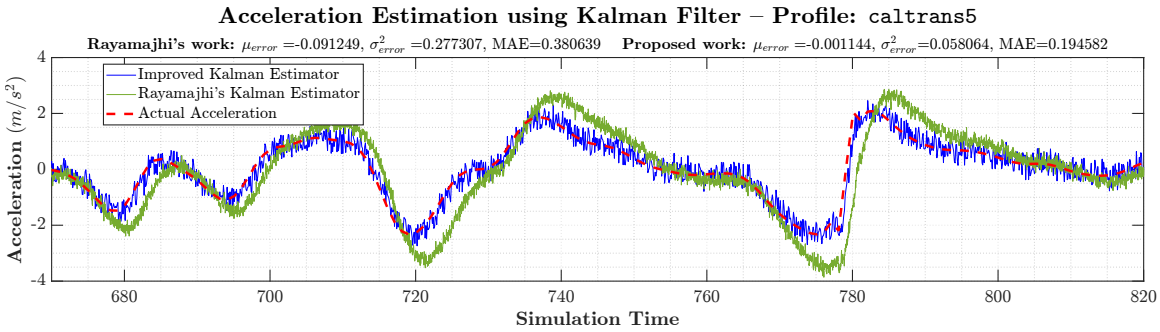


Figure 15: Improvement in acceleration estimation using `sample5` (`caltrans5`) acceleration profile

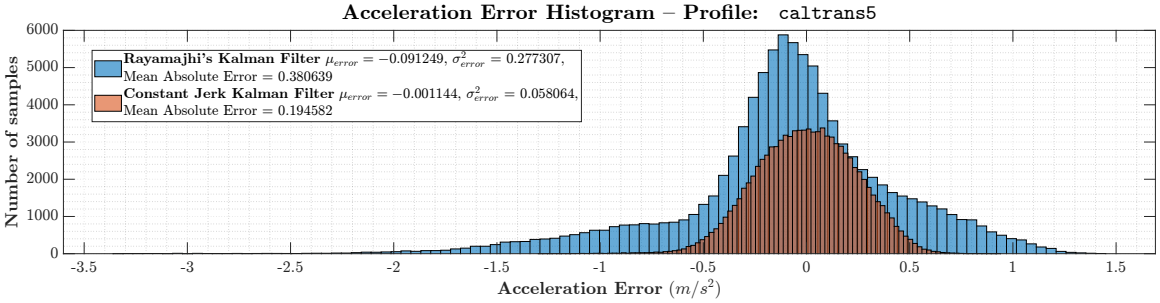


Figure 16: Histogram of acceleration estimation errors for `sample5` (`caltrans5`) acceleration profile

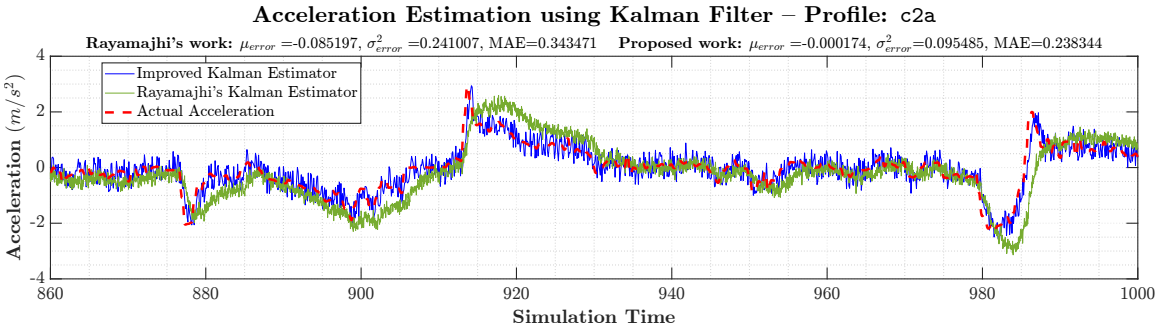


Figure 17: Improvement in acceleration estimation using `c2a` acceleration profile

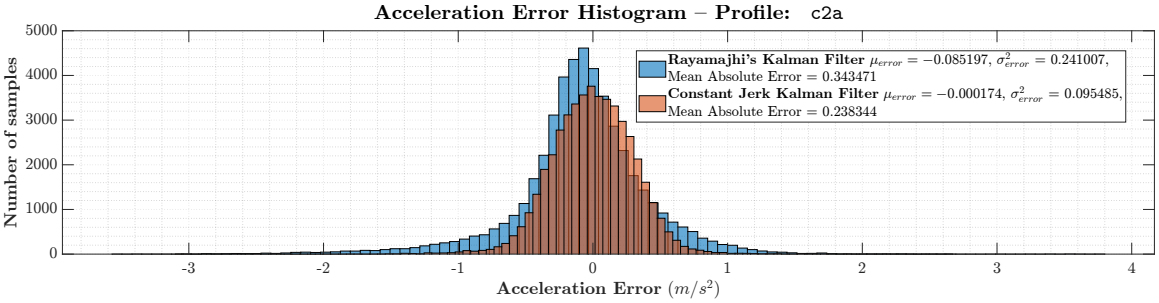


Figure 18: Histogram of acceleration estimation errors for `c2a` acceleration profile

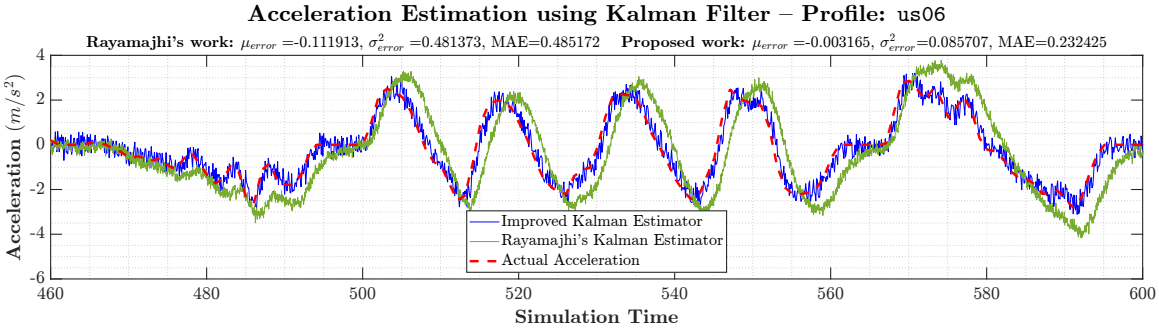


Figure 19: Improvement in acceleration estimation using `us06` acceleration profile

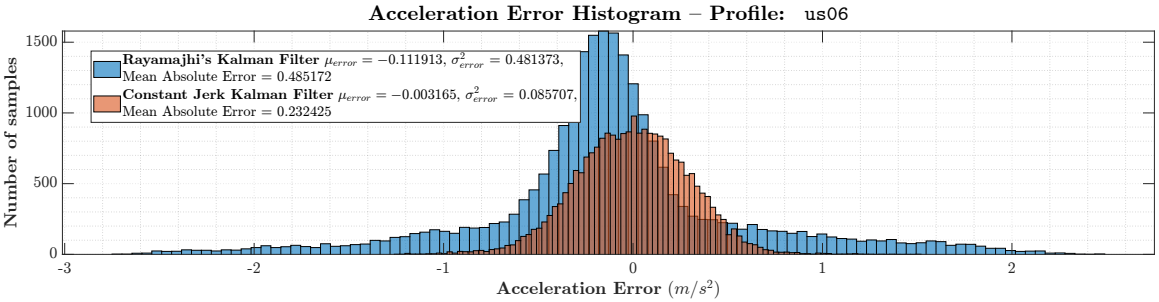


Figure 20: Histogram of acceleration estimation errors for `us06` acceleration profile

Appendix D Improved distance estimation using Kalman filter

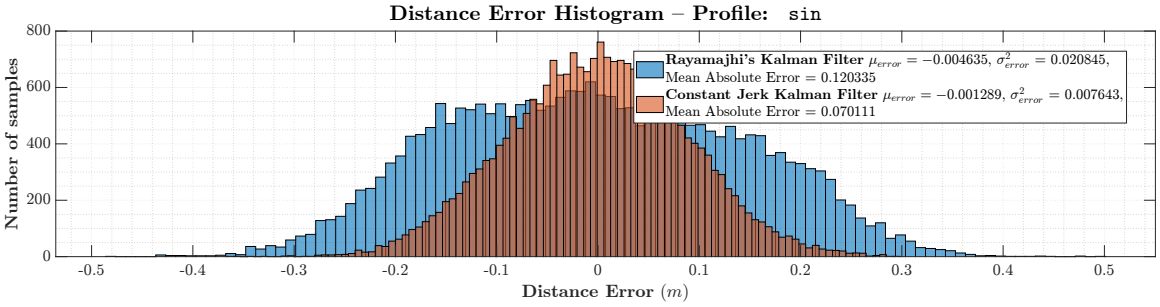


Figure 21: Distance error histogram. Linear profile

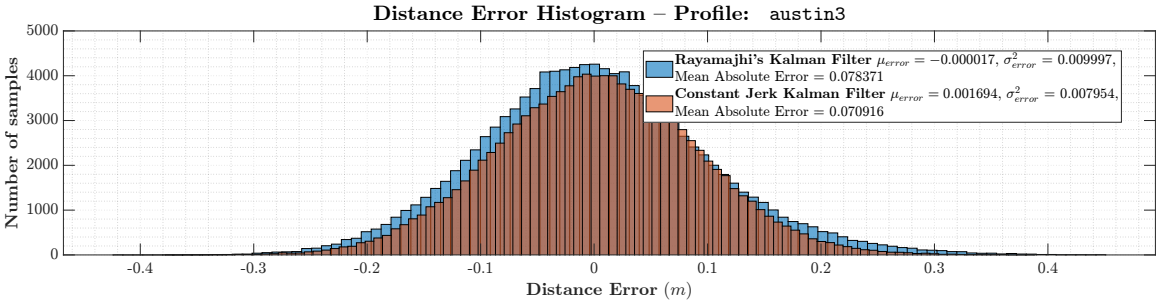


Figure 22: Distance error histogram. `austin3` profile

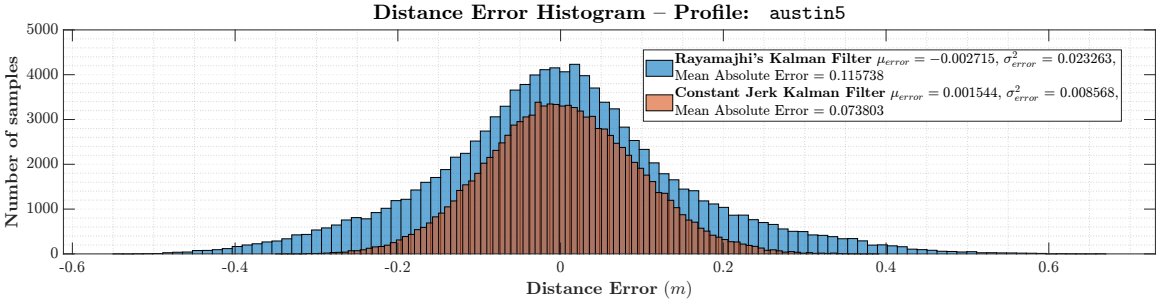


Figure 23: Distance error histogram. `austin5` profile

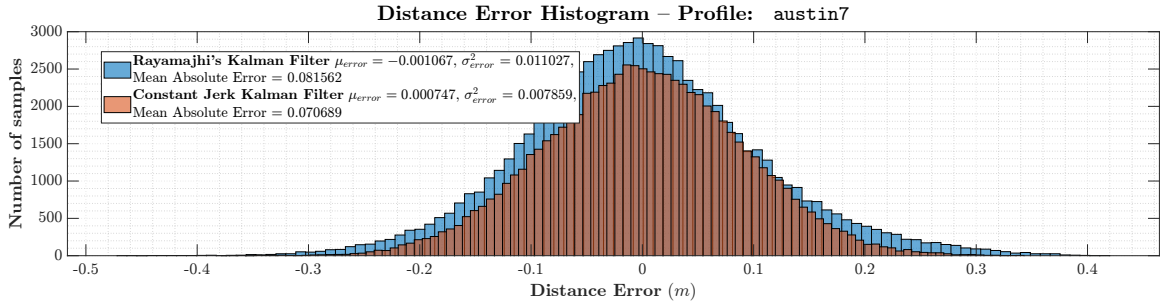


Figure 24: Distance error histogram. `austin7` profile

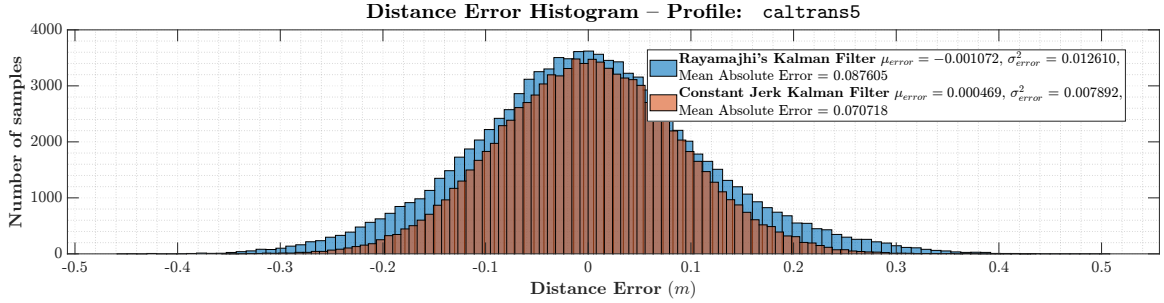


Figure 25: Distance error histogram. `caltrans5` profile

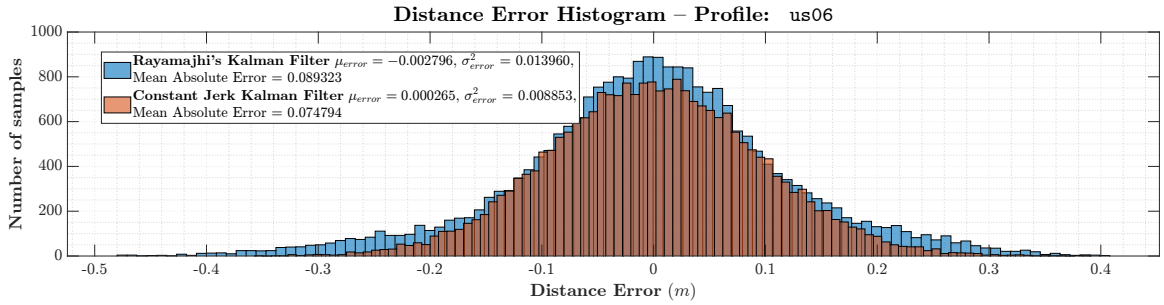


Figure 26: Distance error histogram. `us06` profile

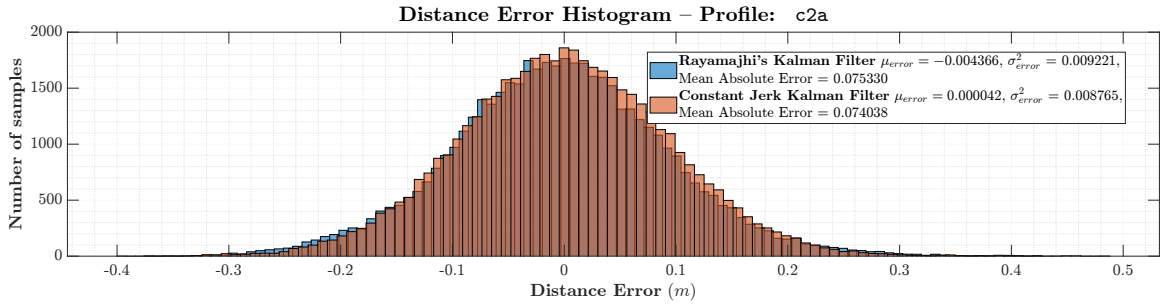
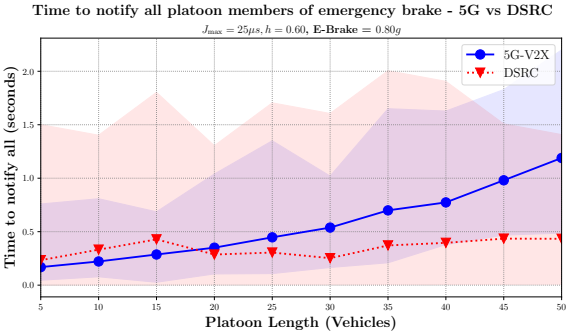


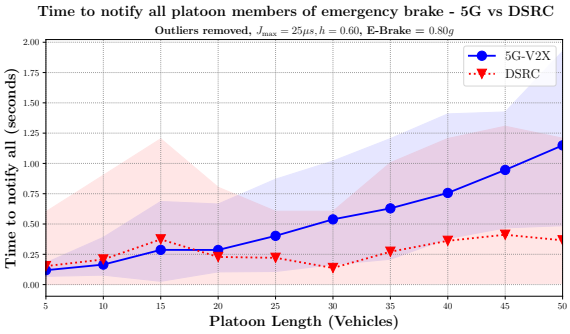
Figure 27: Distance error histogram. `c2a` profile

Appendix E DSRC versus 5G NR V2X: Latency results with Emergency brake $= 0.8g$

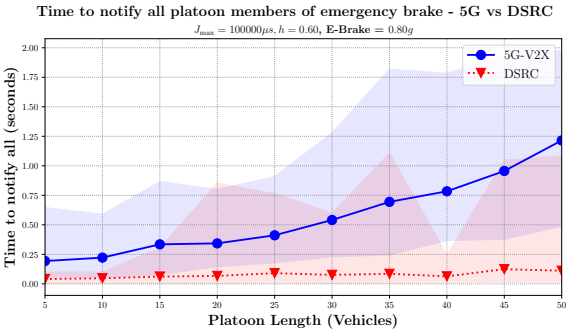
E.1 Time headway $h = 0.6$



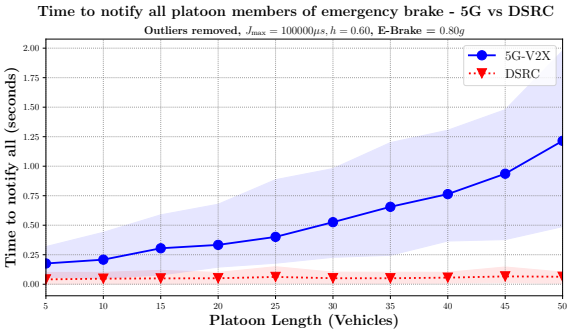
(a) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.8g$



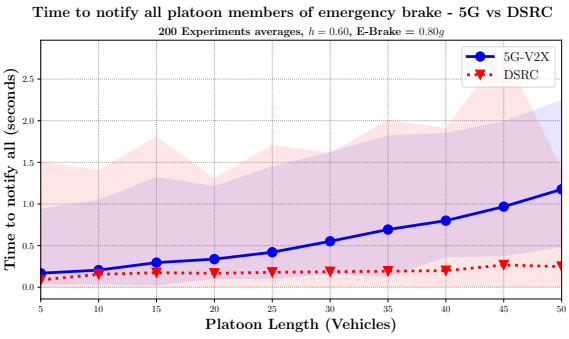
(b) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.8g$, Outliers removed



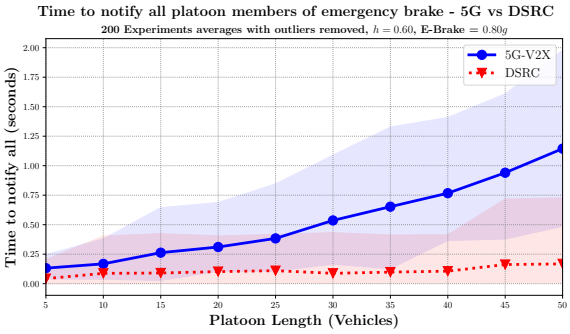
(c) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.8g$, Outliers removed



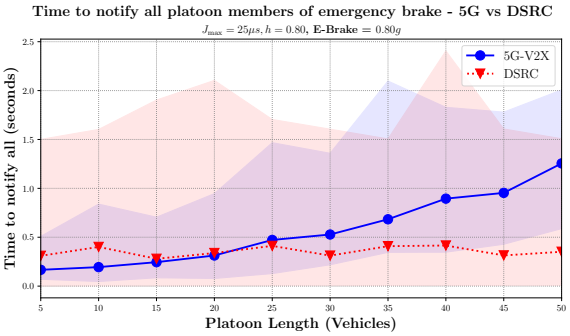
(e) e-brake $= 0.8g, h = 0.6$, 200 Experiments average



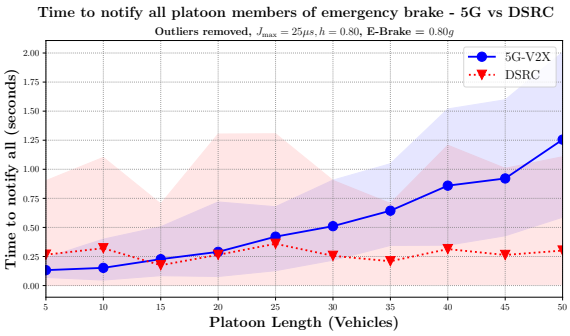
(f) e-brake $= 0.8g, h = 0.6$, 200 Experiments average with outliers removed

Figure 28: Time required to notify all platoon members of emergency using $h = 0.6$, emergency brake $= 0.8g$

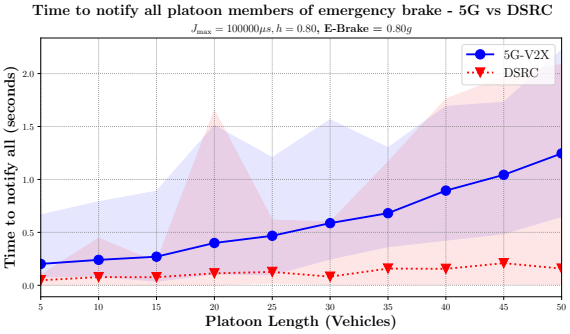
E.2 Time headway $h = 0.8$



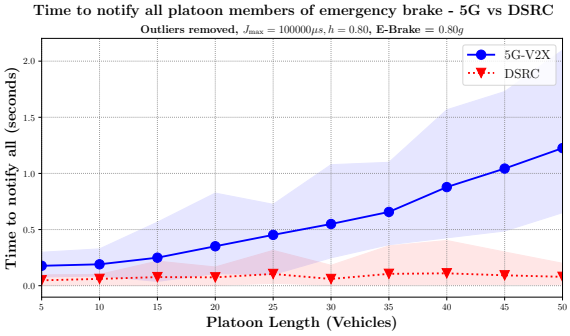
(a) $J_{max} = 25\mu s, h = 0.8$, e-brake $= 0.8g$



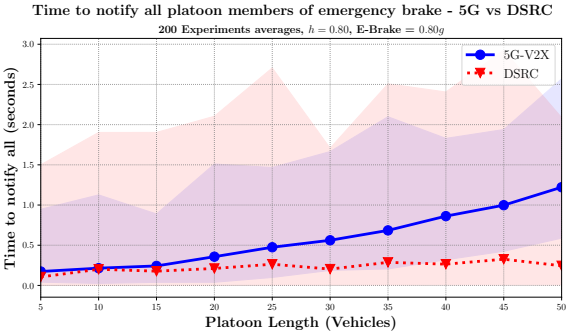
(b) $J_{max} = 25\mu s$, e-brake $= 0.8g, h = 0.8$, Outliers removed



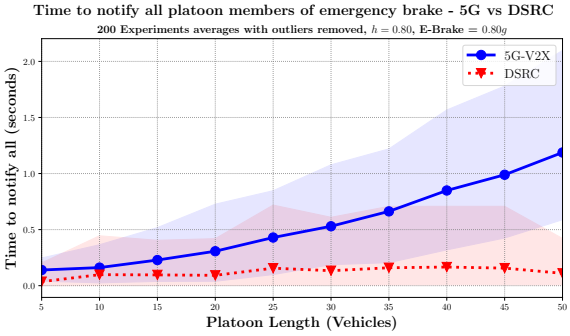
(c) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.8g$, Outliers removed



(e) e-brake $= 0.8g, h = 0.8$, 200 Experiments average

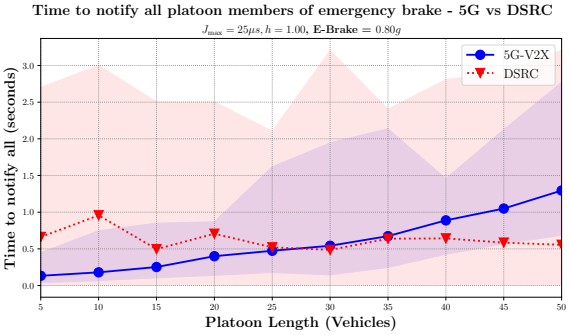


(f) e

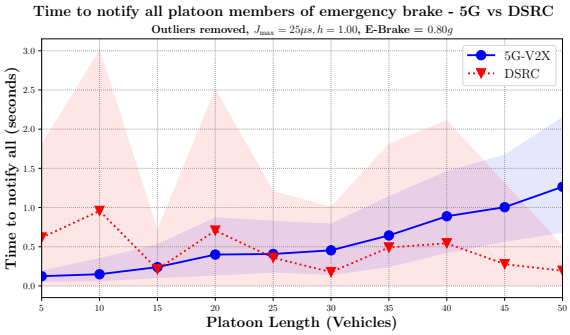
-brake $= 0.8g, h = 0.8$, 200 Experiments average with outliers removed

Figure 29: Time required to notify all platoon members of emergency using $h = 0.8$, emergency brake $= 0.8g$

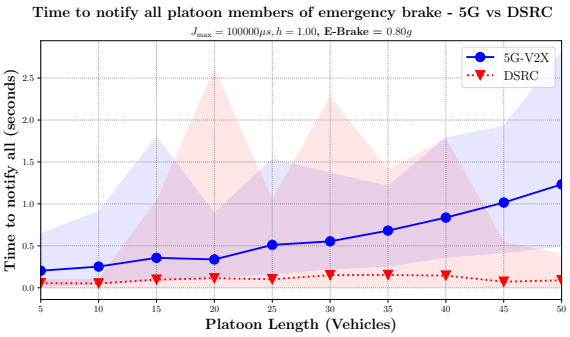
E.3 Time headway $h = 1.0$



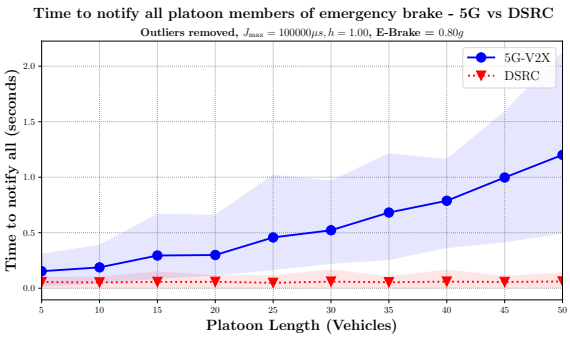
(a) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.8g$



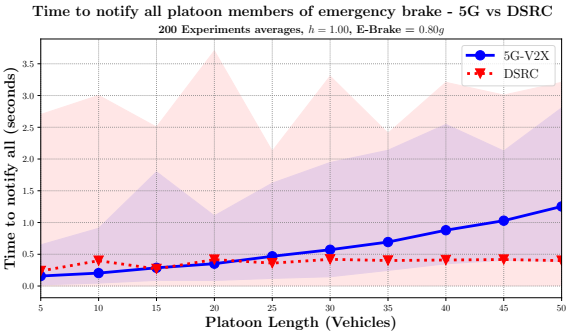
(b) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.8g$, Outliers removed



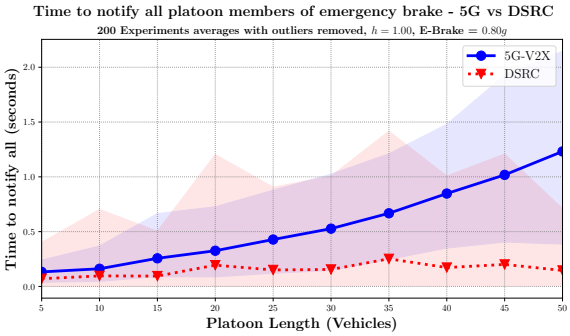
(c) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.8g$, Outliers removed



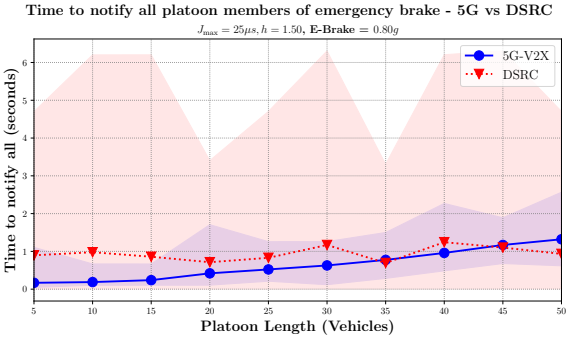
(e) 200 Experiments average, e-brake $= 0.8g, h = 1.0$



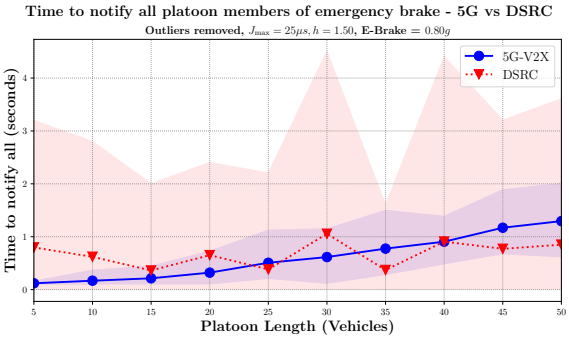
(f) e-brake $= 0.8g, h = 1.0$, 200 Experiments average with outliers removed

Figure 30: Time required to notify all platoon members of emergency using $h = 1.0$, emergency brake $= 0.8g$

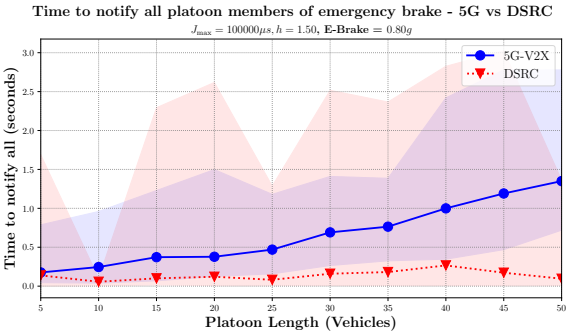
E.4 Time headway $h = 1.5$



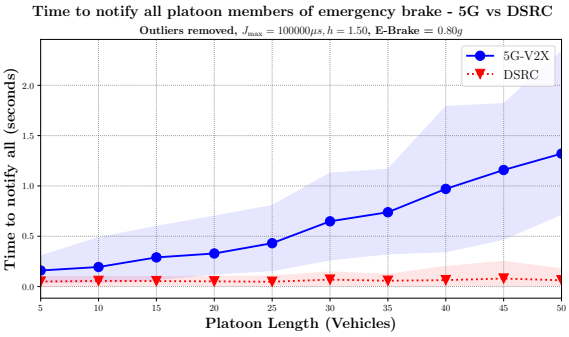
(a) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.8g$



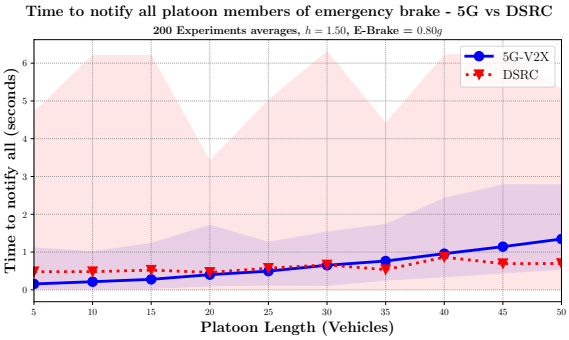
(b) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.8g$, Outliers removed



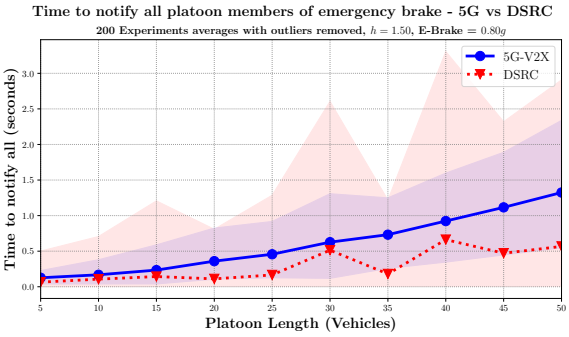
(c) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.8g$, Outliers removed



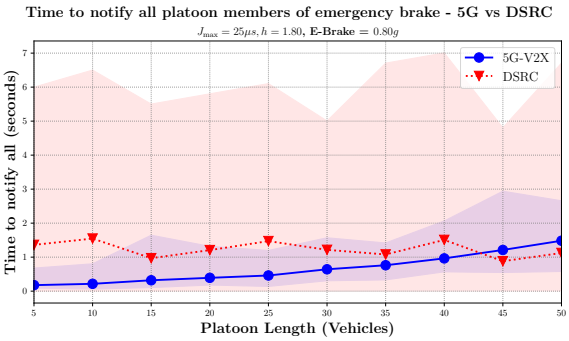
(e) 200 Experiments average, e-brake $= 0.8g, h = 1.5$



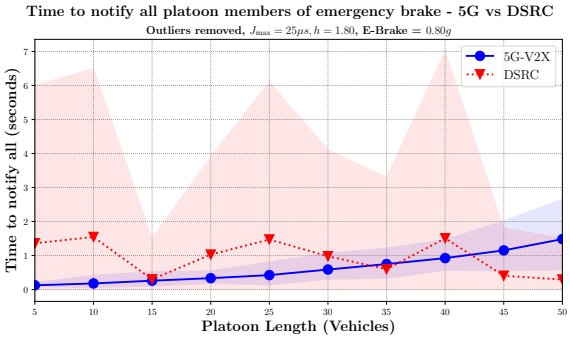
(f) e-brake $= 0.8g, h = 1.5$, 200 Experiments average with outliers removed

Figure 31: Time required to notify all platoon members of emergency using $h = 1.5$, emergency brake $= 0.8g$

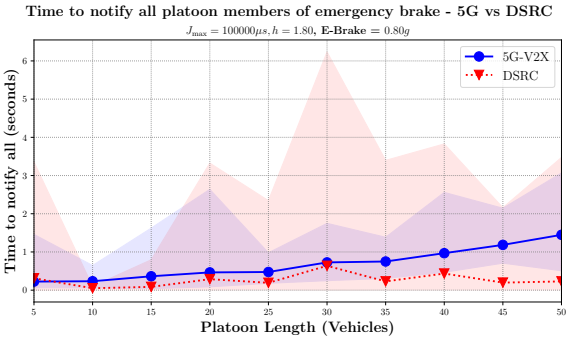
E.5 Time headway $h = 1.8$



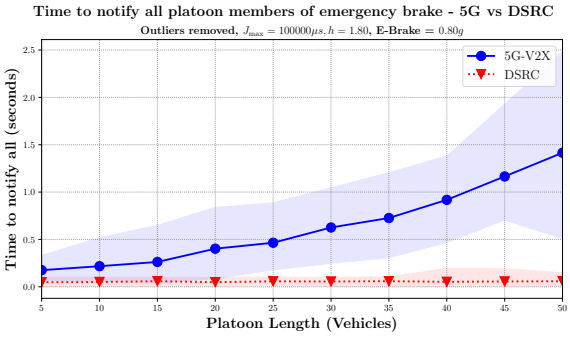
(a) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.8g$



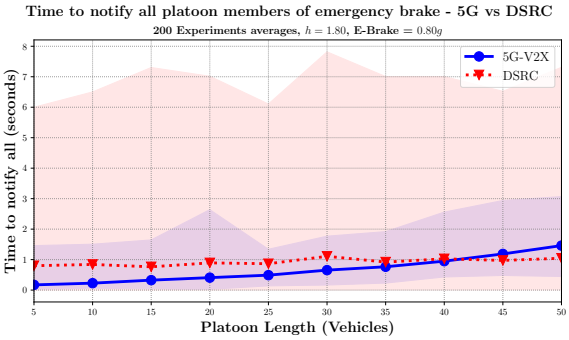
(b) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.8g$, Outliers removed



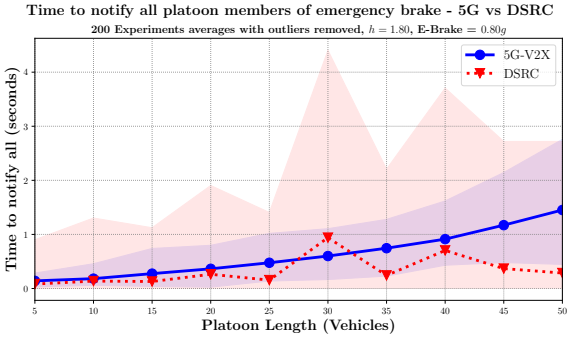
(c) $J_{max} = 100000\mu s, h = 1.8$, e-brake $= 0.8g$



(d) Sample 2 Acceleration



(e) 200 Experiments average, e-brake $= 0.8g, h = 1.8$

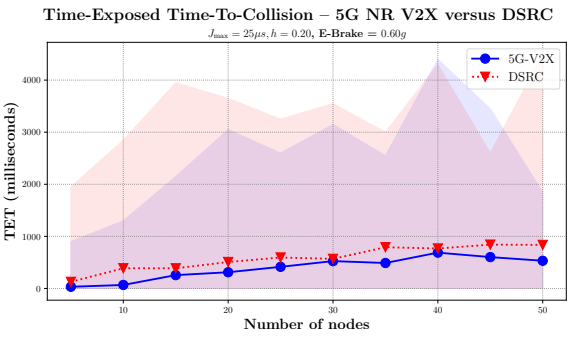


(f) e-brake $= 0.8g, h = 1.8$, 200 Experiments average with outliers removed

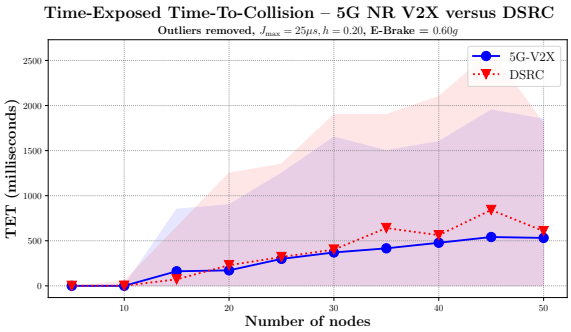
Figure 32: Time required to notify all platoon members of emergency using $h = 1.8$, emergency brake $= 0.8g$

Appendix F DSRC versus 5G NR V2X: Time-Exposed Time-to-collision (TET)

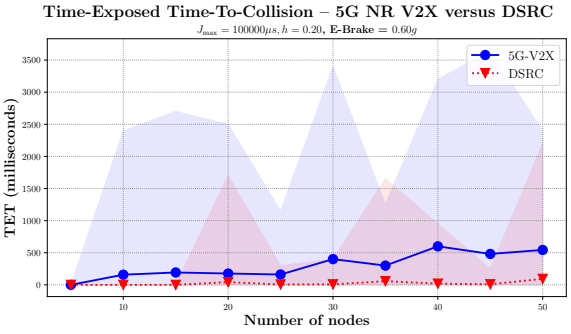
F.1 Time headway $h = 0.2$, Emergency brake $= 0.6g$



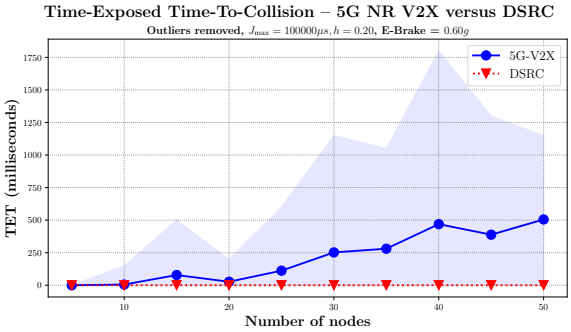
(a) $J_{max} = 25\mu s, h = 0.2$, e-brake $= 0.6g$



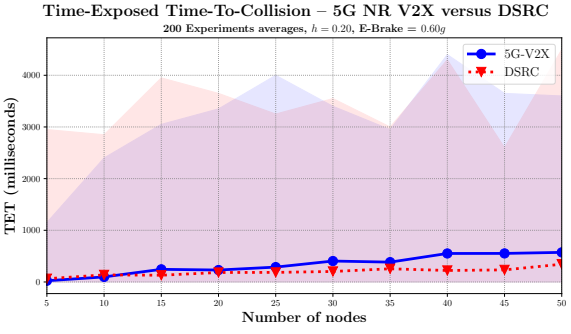
(b) $J_{max} = 25\mu s, h = 0.2$, e-brake $= 0.6g$, Outliers removed



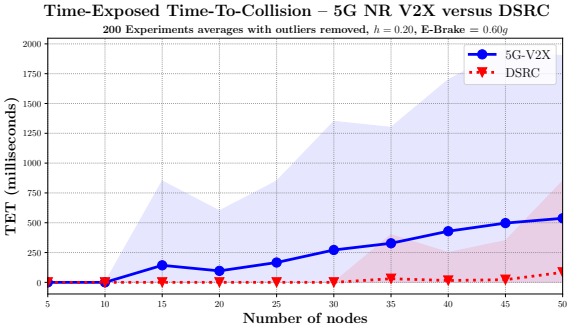
(c) $J_{max} = 100000\mu s, h = 0.2$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 0.2$, e-brake $= 0.6g$, Outliers removed



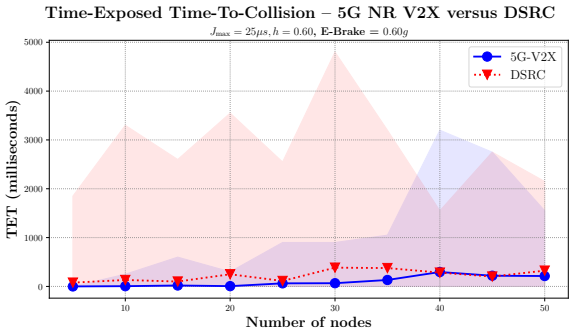
(e) 200 Experiments average, e-brake $= 0.6g, h = 0.2$



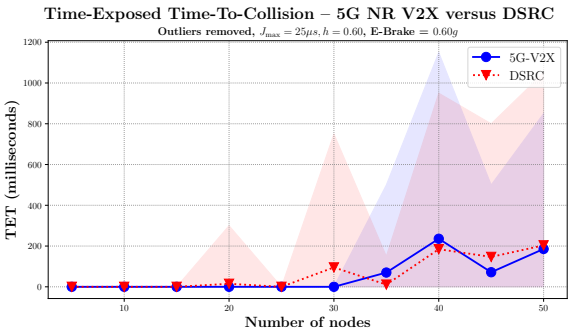
(f) e-brake $= 0.6g, h = 0.2$, 200 Experiments average outliers removed

Figure 33: Time-Exposed Time-to-collision (TET) using $h = 0.2$, emergency brake $= 0.6g$

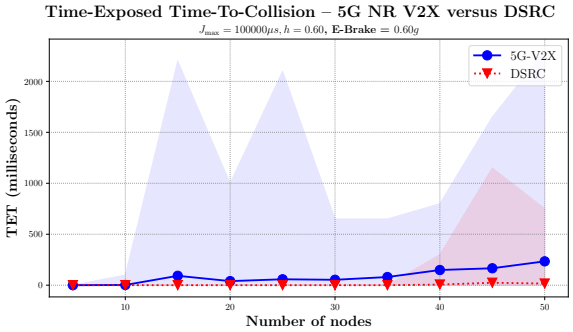
F.2 Time headway $h = 0.6$, Emergency brake $= 0.6g$



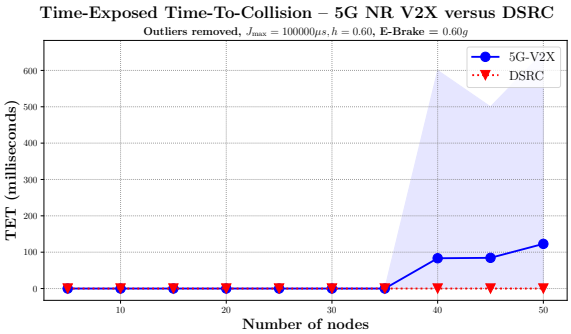
(a) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.6g$



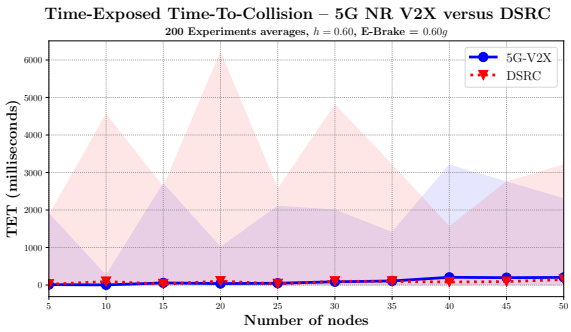
(b) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.6g$, Outliers removed



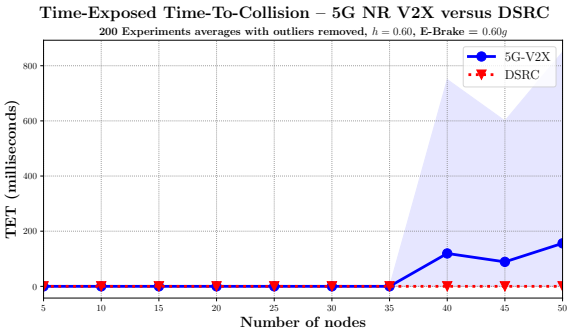
(c) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.6g$, Outliers removed



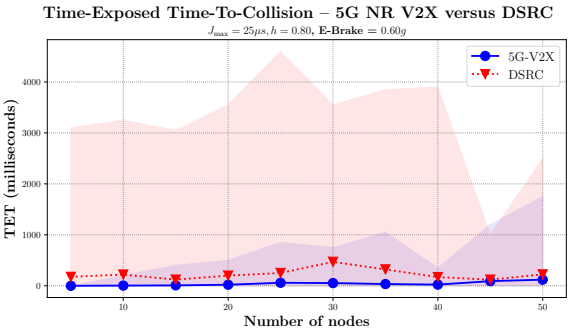
(e) 200 Experiments average, e-brake $= 0.6g, h = 0.6$



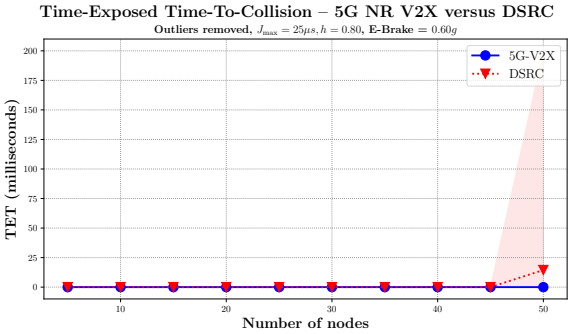
(f) e-brake $= 0.6g, h = 0.6$, 200 Experiments average with outliers removed

Figure 34: Time-Exposed Time-to-collision (TET) using $h = 0.6$, emergency brake $= 0.6g$

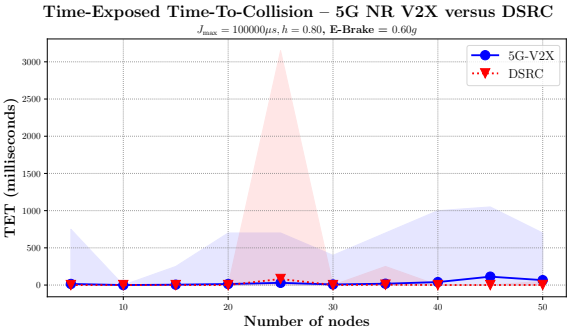
F.3 Time headway $h = 0.8$, Emergency brake $= 0.6g$



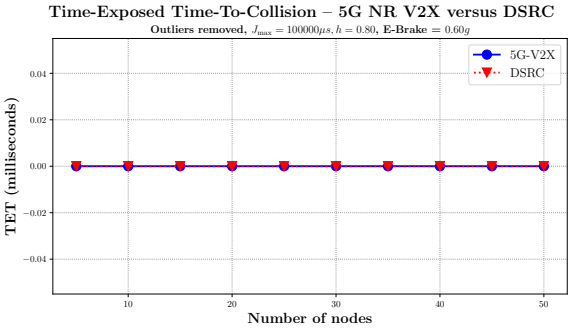
(a) $J_{max} = 25\mu s, h = 0.8$, e-brake $= 0.6g$



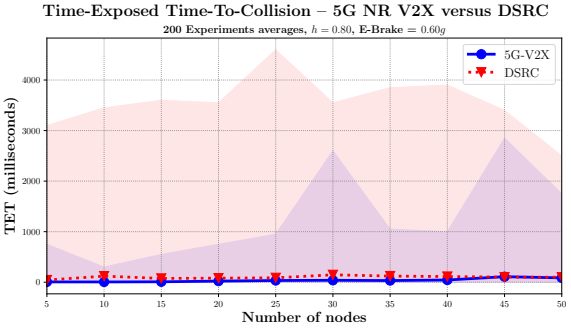
(b) $J_{max} = 25\mu s, h = 0.8$, e-brake $= 0.6g$, Outliers removed



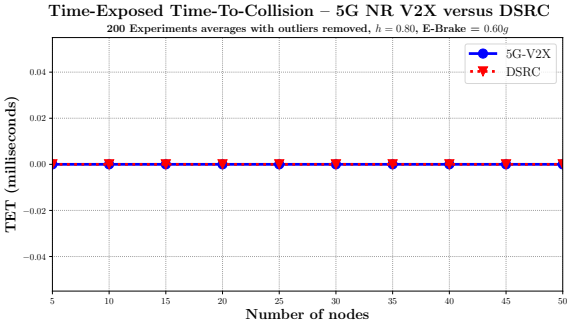
(c) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.6g$, Outliers removed



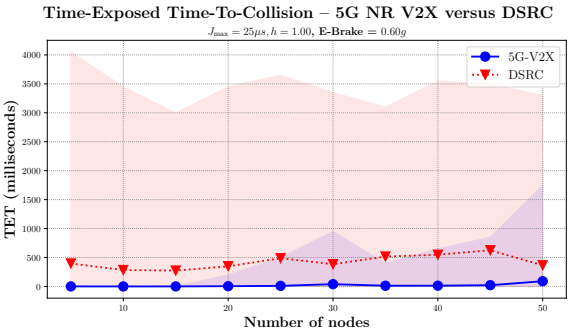
(e) 200 Experiments average, e-brake $= 0.6g, h = 0.8$



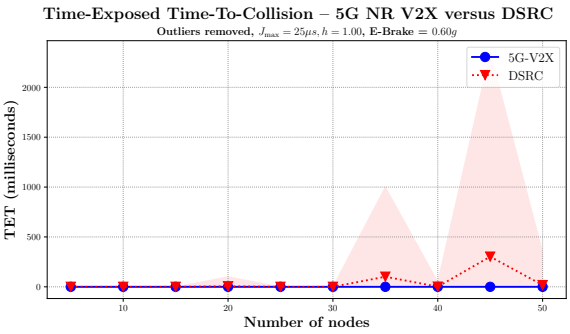
(f) e-brake $= 0.6g, h = 0.8$, 200 Experiments average with outliers removed

Figure 35: Time-Exposed Time-to-collision (TET) using $h = 0.8$, emergency brake $= 0.6g$

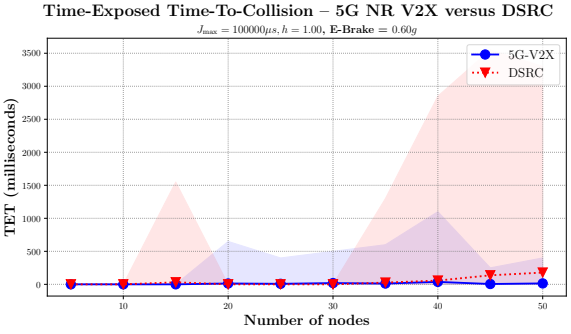
F.4 Time headway $h = 1.0$, Emergency brake $= 0.6g$



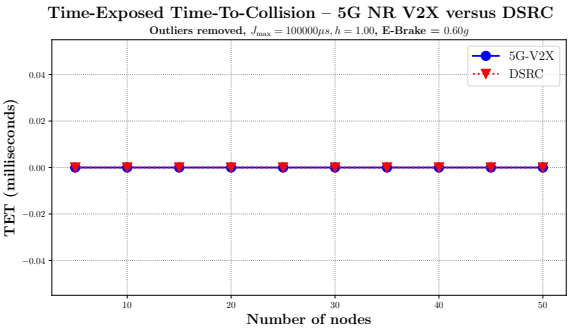
(a) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.6g$



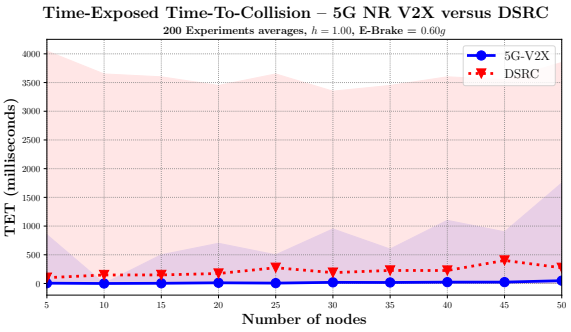
(b) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.6g$, Outliers removed



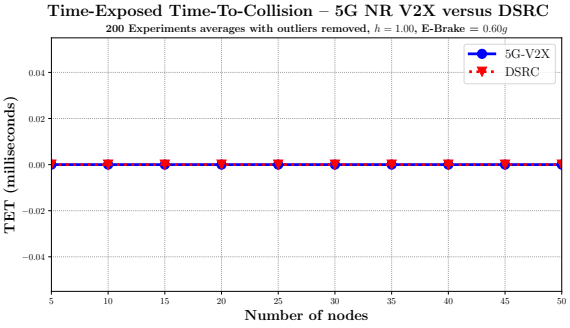
(c) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.6g$, Outliers removed



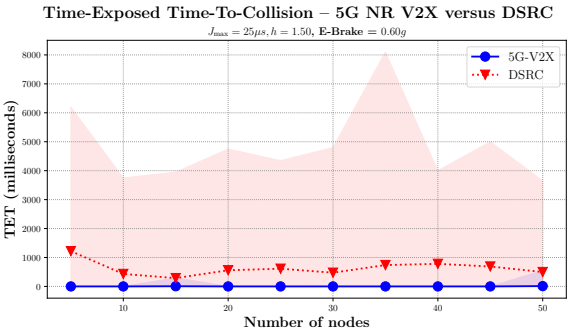
(e) 200 Experiments average, e-brake $= 0.6g, h = 1.0$



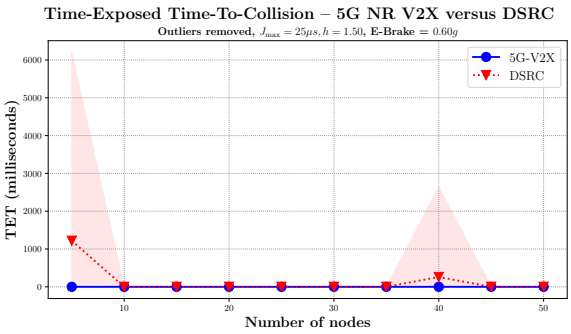
(f) e-brake $= 0.6g, h = 1.0$, 200 Experiments average with outliers removed

Figure 36: Time-Exposed Time-to-collision (TET) using $h = 1.0$, emergency brake $= 0.6g$

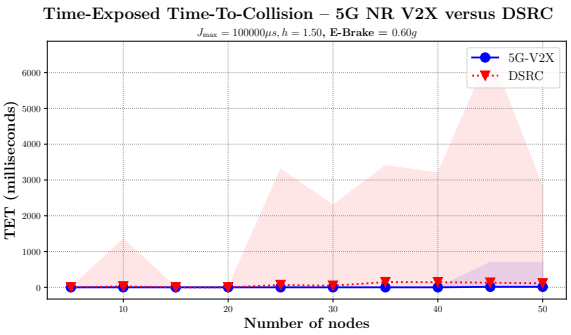
F.5 Time headway $h = 1.5$, Emergency brake $= 0.6g$



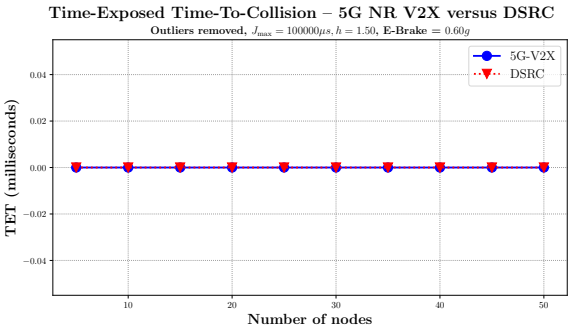
(a) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.6g$



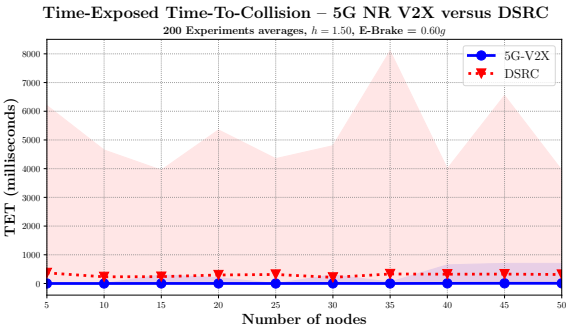
(b) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.6g$, Outliers removed



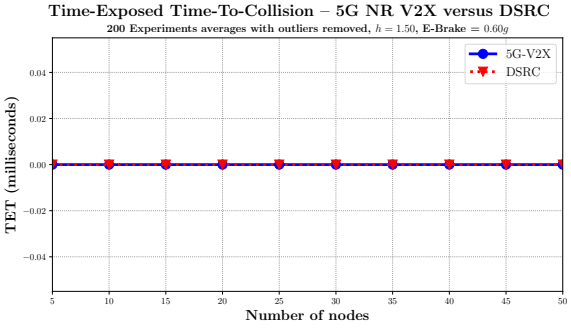
(c) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.6g$, Outliers removed



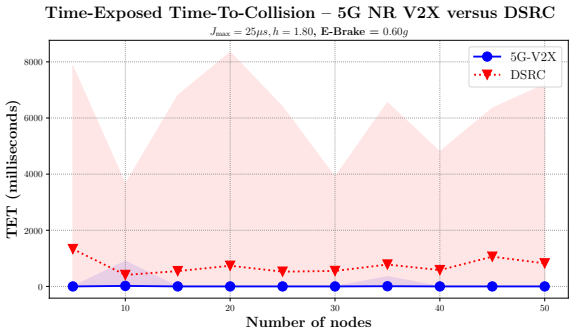
(e) 200 Experiments average, e-brake $= 0.6g, h = 1.5$



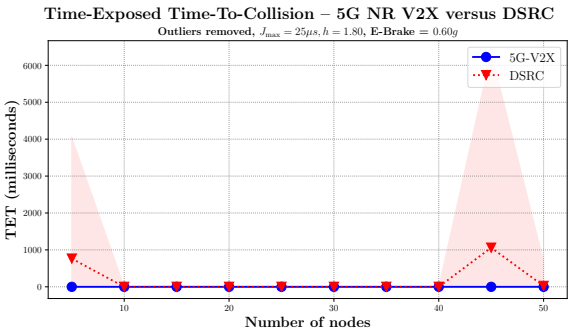
(f) e-brake $= 0.6g, h = 1.5$, 200 Experiments average with outliers removed

Figure 37: Time-Exposed Time-to-collision (TET) using $h = 1.5$, emergency brake $= 0.6g$

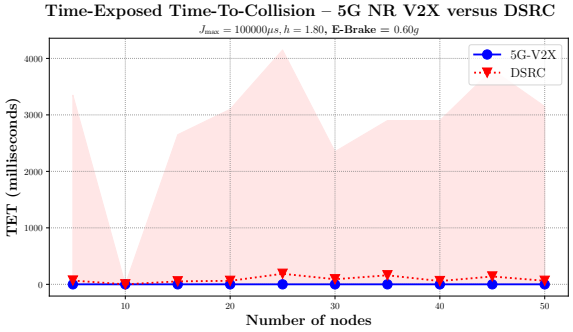
F.6 Time headway $h = 1.8$, Emergency brake $= 0.6g$



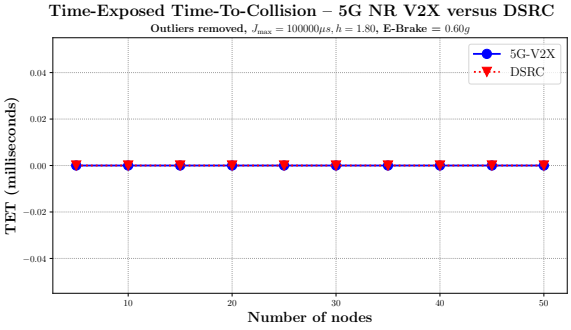
(a) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.6g$



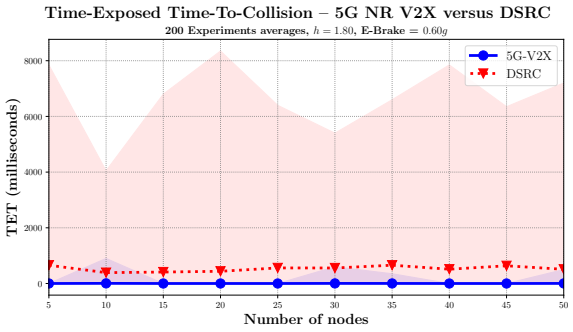
(b) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.6g$, Outliers removed



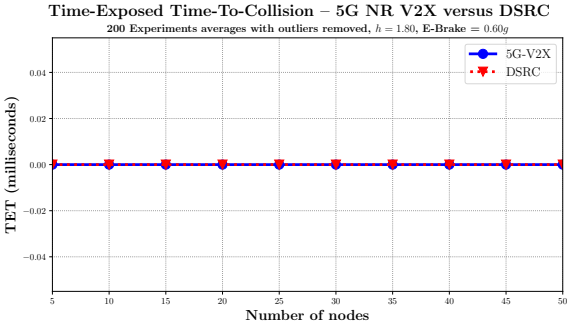
(c) $J_{max} = 100000\mu s, h = 1.8$, e-brake $= 0.6g$



(d) $J_{max} = 100000\mu s, h = 1.8$, e-brake $= 0.6g$, Outliers removed



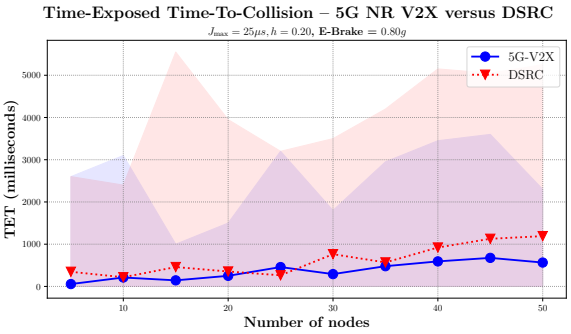
(e) 200 Experiments average, e-brake $= 0.6g, h = 1.8$



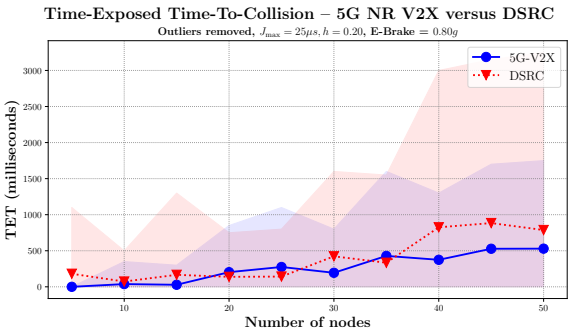
(f) e-brake $= 0.6g, h = 1.8$, 200 Experiments average with outliers removed

Figure 38: Time-Exposed Time-to-collision (TET) using $h = 1.8$, emergency brake $= 0.6g$

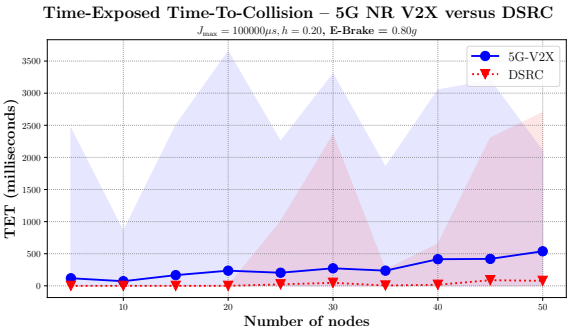
F.7 Time headway $h = 0.2$, Emergency brake $= 0.8g$



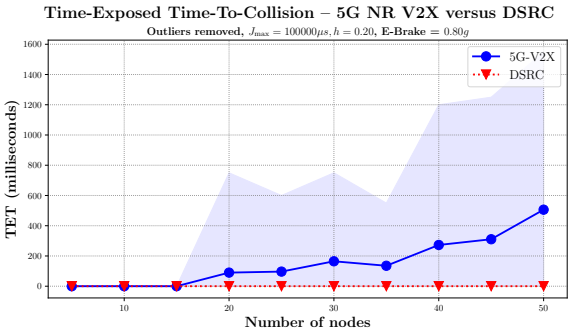
(a) $J_{max} = 25\mu s, h = 0.2$, e-brake $= 0.8g$



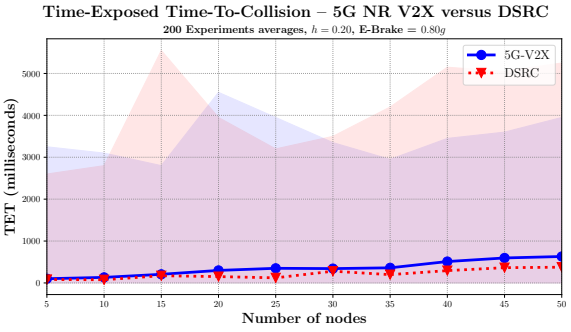
(b) $J_{max} = 25\mu s, h = 0.2$, e-brake $= 0.8g$, Outliers removed



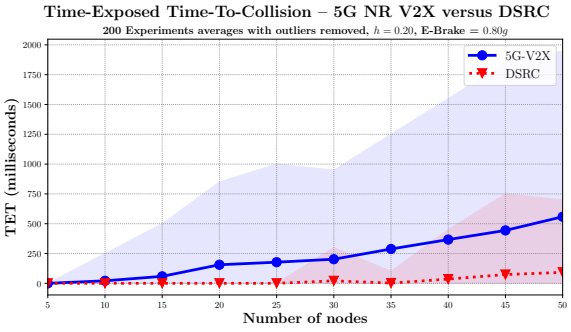
(c) $J_{max} = 100000\mu s, h = 0.2$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 0.2$, e-brake $= 0.8g$, Outliers removed



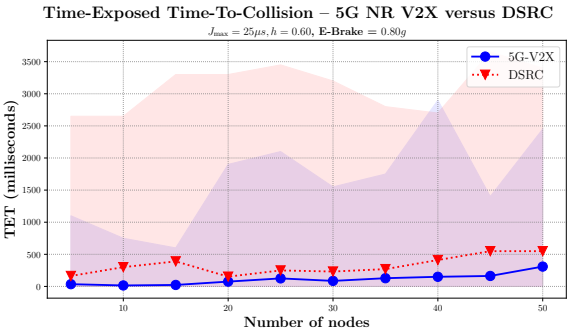
(e) 200 Experiments average, e-brake $= 0.8g, h = 0.2$



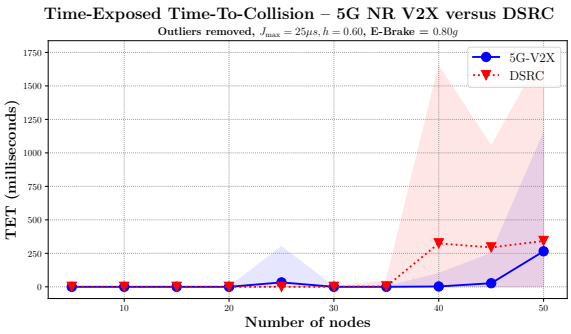
(f) e-brake $= 0.8g, h = 0.2$, 200 Experiments average with outliers removed

Figure 39: Time-Exposed Time-to-collision (TET) using $h = 0.2$, emergency brake $= 0.8g$

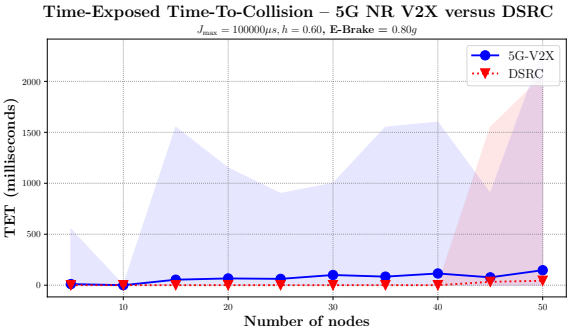
F.8 Time headway $h = 0.6$, Emergency brake $= 0.8g$



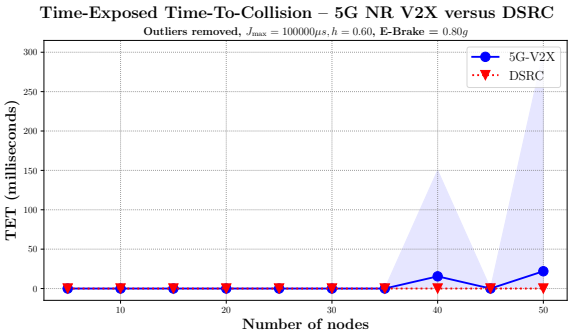
(a) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.8g$



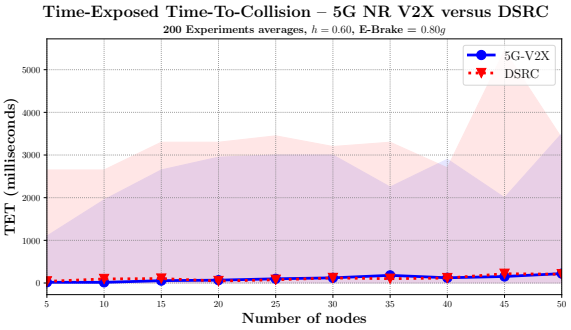
(b) $J_{max} = 25\mu s, h = 0.6$, e-brake $= 0.8g$, Outliers removed



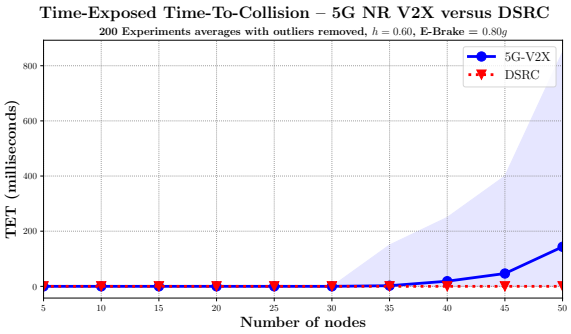
(c) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 0.6$, e-brake $= 0.8g$, Outliers removed



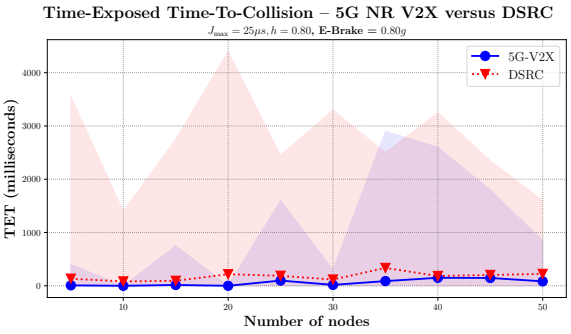
(e) 200 Experiments average, e-brake $= 0.8g, h = 0.6$



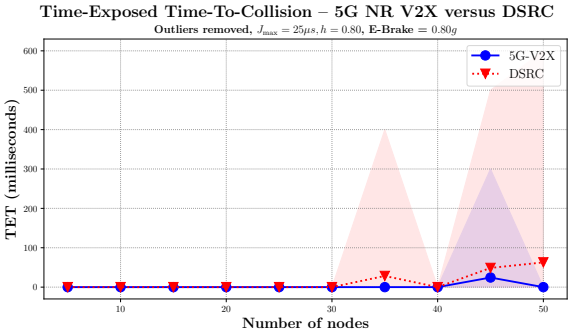
(f) e-brake $= 0.8g, h = 0.6$, 200 Experiments average with outliers removed

Figure 40: Time-Exposed Time-to-collision (TET) using $h = 0.6$, emergency brake $= 0.8g$

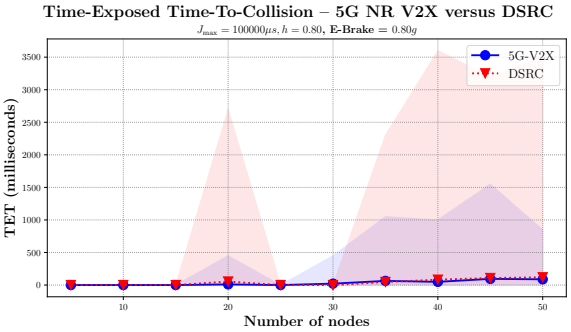
F.9 Time headway $h = 0.8$, Emergency brake $= 0.8g$



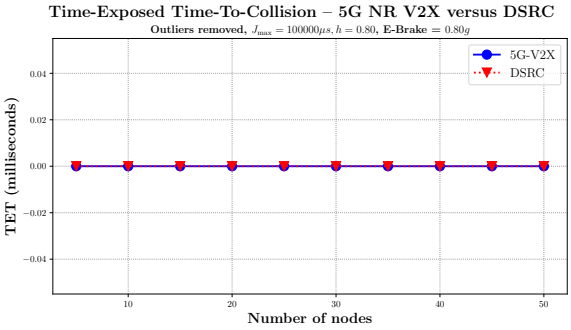
(a) $J_{max} = 25\mu s, h = 0.8$, e-brake $= 0.8g$



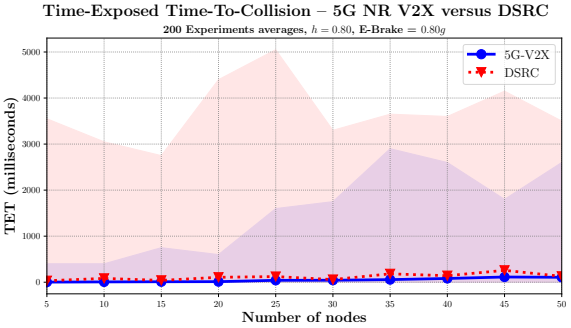
(b) $J_{max} = 25\mu s, h = 0.8$, e-brake $= 0.8g$, Outliers removed



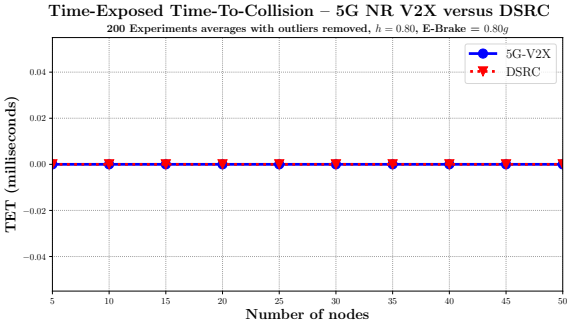
(c) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 0.8$, e-brake $= 0.8g$, Outliers removed



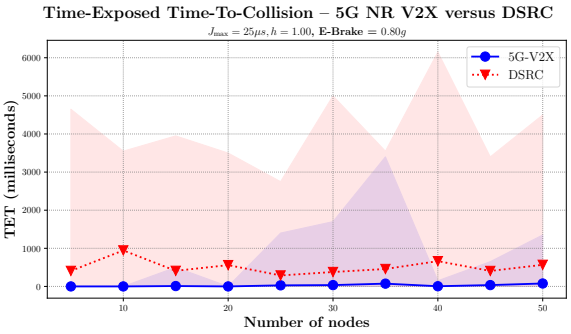
(e) 200 Experiments average, e-brake $= 0.8g, h = 0.8$



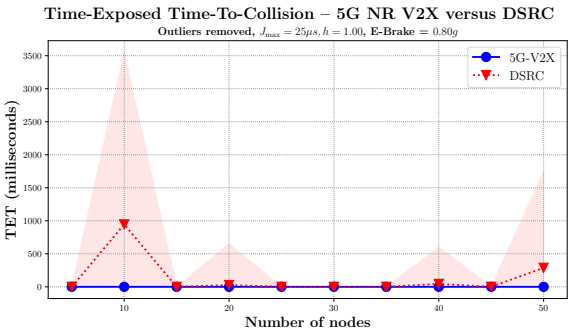
(f) e-brake $= 0.8g, h = 0.8$, 200 Experiments average with outliers removed

Figure 41: Time-Exposed Time-to-collision (TET) using $h = 0.8$, emergency brake $= 0.8g$

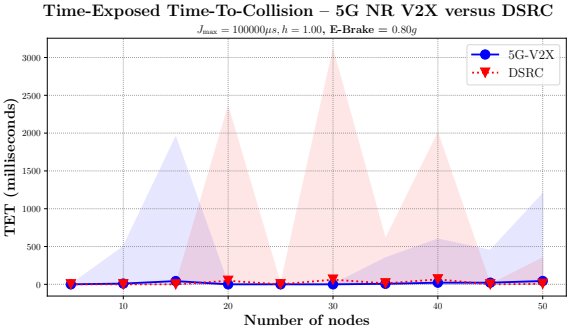
F.10 Time headway $h = 1.0$, Emergency brake $= 0.8g$



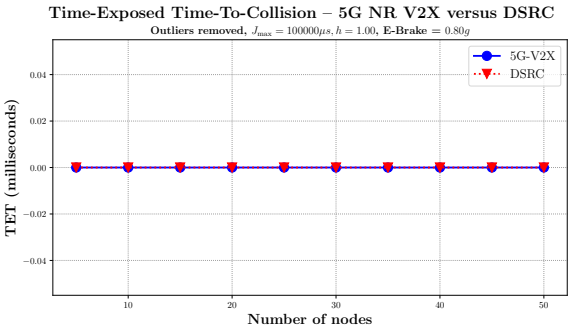
(a) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.8g$



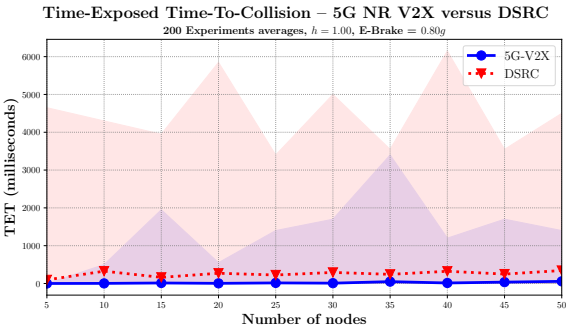
(b) $J_{max} = 25\mu s, h = 1.0$, e-brake $= 0.8g$, Outliers removed



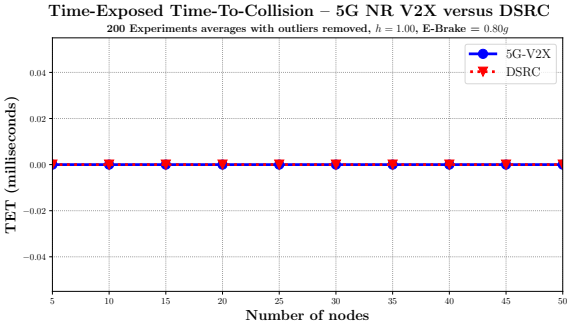
(c) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 1.0$, e-brake $= 0.8g$, Outliers removed



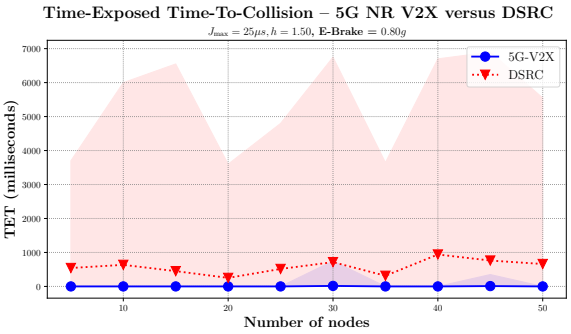
(e) 200 Experiments average, e-brake $= 0.8g, h = 1.0$



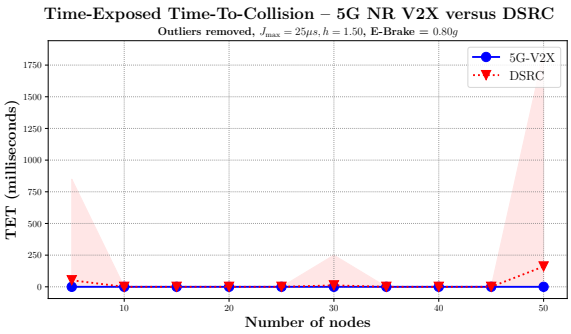
(f) e-brake $= 0.8g, h = 1.0$, 200 Experiments average with outliers removed

Figure 42: Time-Exposed Time-to-collision (TET) using $h = 1.0$, emergency brake $= 0.8g$

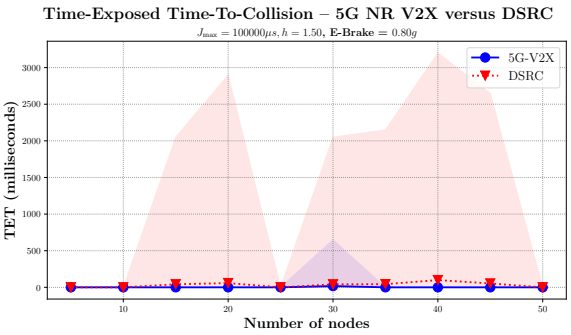
F.11 Time headway $h = 1.5$, Emergency brake $= 0.8g$



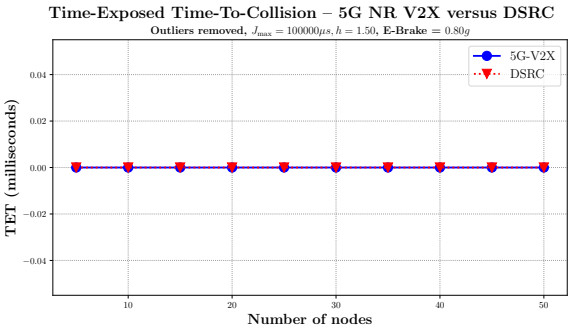
(a) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.8g$



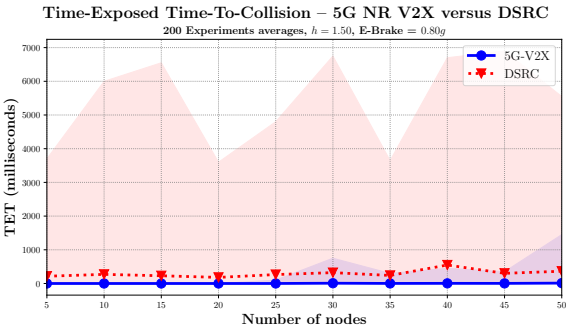
(b) $J_{max} = 25\mu s, h = 1.5$, e-brake $= 0.8g$, Outliers removed



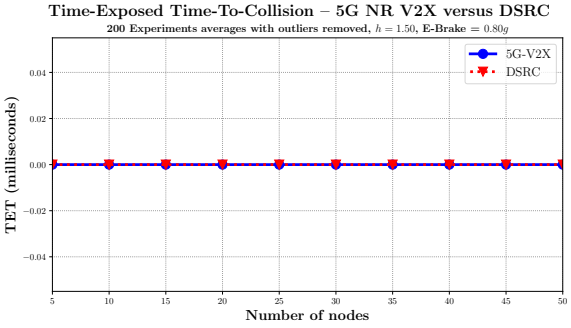
(c) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 1.5$, e-brake $= 0.8g$, Outliers removed



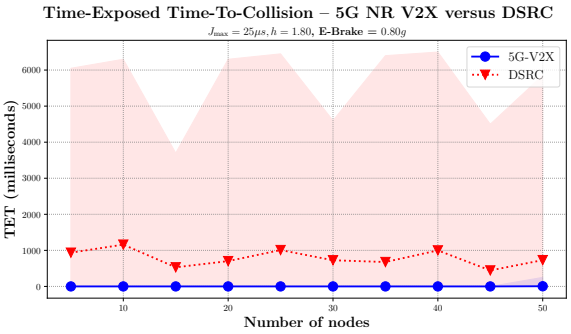
(e) 200 Experiments average, e-brake $= 0.8g, h = 1.5$



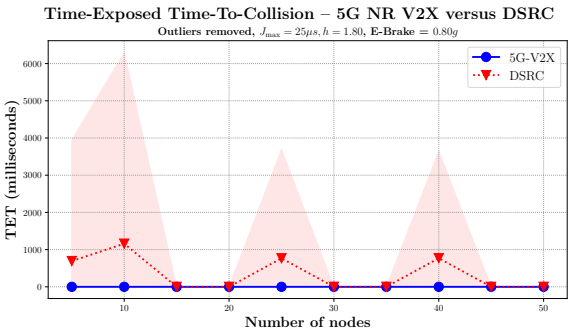
(f) e-brake $= 0.8g, h = 1.5$, 200 Experiments average with outliers removed

Figure 43: Time-Exposed Time-to-collision (TET) using $h = 1.5$, emergency brake $= 0.8g$

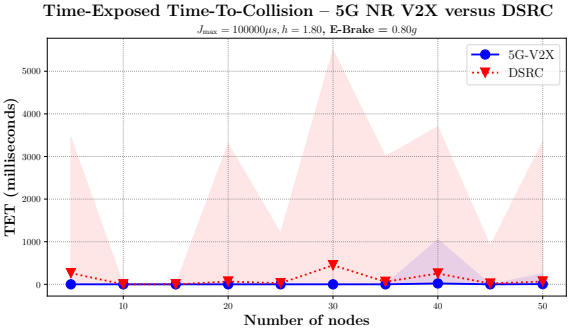
F.12 Time headway $h = 1.8$, Emergency brake $= 0.8g$



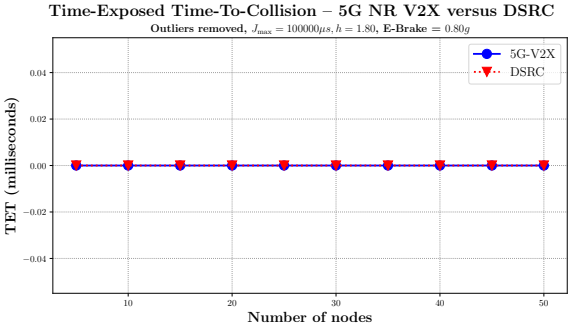
(a) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.8g$



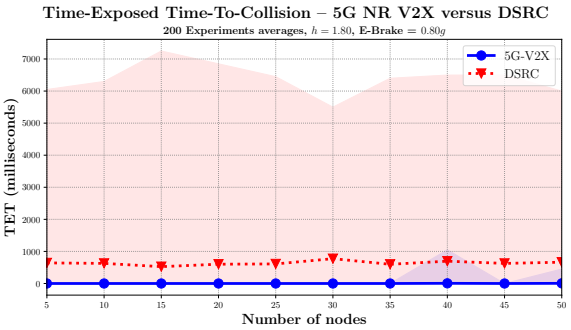
(b) $J_{max} = 25\mu s, h = 1.8$, e-brake $= 0.8g$, Outliers removed



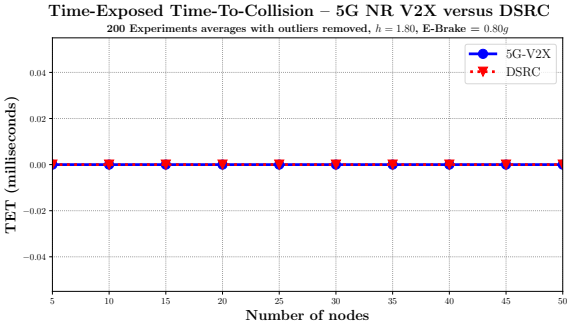
(c) $J_{max} = 100000\mu s, h = 1.8$, e-brake $= 0.8g$



(d) $J_{max} = 100000\mu s, h = 1.8$, e-brake $= 0.8g$, Outliers removed



(e) 200 Experiments average, e-brake $= 0.8g, h = 1.8$



(f) e-brake $= 0.8g, h = 1.8$, 200 Experiments average with outliers removed

Figure 44: Time-Exposed Time-to-collision (TET) using $h = 1.8$, emergency brake $= 0.8g$

Bibliography

[1] National Renewable Energy Laboratory (NREL), Transportation Secure Data Center - Drive Cycle Data. https://www.nrel.gov/transportation/secure-transportation-data/tsdc-drive-cycle-data.html. Accessed: 2020-10-14.

[2] IEEE standard for information technology – Telecommunications and information exchange between systems local and metropolitan area networks—specific requirements - part 11: Wireless lan medium access control (MAC) and physical layer (PHY) specifications. *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pages 1–3534, Dec. 2016.

[3] P802.11bd - Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Enhancements for Next Generation V2X, 2020.

[4] 3GPP. Release 14 description; summary of rel-14 work items. Technical Report TR 21.914, 3rd generation partnership project, 2017.

[5] 3GPP. Release 15 description; summary of rel-15 work items. Technical Report TR 21.915, 3rd generation partnership project, 2018.

[6] 3GPP. Release 16 description; summary of rel-16 work items. Technical Report TR 21.916, 3rd generation partnership project, 2020.

[7] Renato Abreu, Gilberto Berardinelli, Thomas Jacobsen, Klaus Pedersen, and Preben Mogensen. A blind retransmission scheme for ultra-reliable and low latency communications. In *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2018.

[8] Kazi Iftekhar Ahmed. *Modeling drivers' acceleration and lane changing behavior*. PhD thesis, Massachusetts Institute of Technology, 1999.

[9] Nasser H Ali and Ghassan M Hassan. Kalman filter tracking. *International Journal of Computer Applications*, 89(9), 2014.

[10] Zoraze Ali, Sandra Lagén, Lorenza Giupponi, and Richard Rouil. 3gpp nr v2x mode 2: Overview, models and system-level evaluation. *IEEE Access*, 2021.

[11] Hamada Alshaer and Eric Horlait. An optimized adaptive broadcast scheme for inter-vehicle communication. In *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, volume 5, pages 2840–2844. IEEE, 2005.

[12] Adil Alsuhaim, Anjan Rayamajhi, James Westall, and Jim Martin. Adapting time headway in cooperative adaptive cruise control to network reliability. *IEEE Transactions on Vehicular Technology*, 70(12):12691–12702, December 2021.

[13] Waqar Anwar, Norman Franchi, and Gerhard Fettweis. Physical layer evaluation of V2X communications technologies: 5G NR-V2X, LTE-V2X, IEEE 802.11 bd, and IEEE 802.11p. In *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pages 1–7. IEEE, 2019.

[14] Stefano Basagni. Distributed clustering for ad hoc networks. In *Parallel Architectures, Algorithms, and Networks, 1999.(I-SPAN'99) Proceedings. Fourth InternationalSymposium on*, pages 310–315. IEEE, 1999.

[15] N. Benamar, J. Härri, J. Lee, and T. Ernest. Basic support for IPv6 networks operating outside the context of a basic service set over IEEE std 802.11. RFC 8691, Internet Engineering Task Force (IETF), December 2019.

[16] Thomas Blazek and Christoph F Mecklenbräuker. Low complexity SNR-based packet level burst-error model for vehicular ad-hoc networks. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–5. IEEE, 2018.

[17] Irina Bocharova, Boris Kudryashov, Maben Rabi, Nikita Lyamin, Wouter Dankers, Erik Frick, and Alexey Vinel. Characterizing packet losses in vehicular networks. *IEEE Transactions on Vehicular Technology*, 68(9):8347–8358, 2019.

[18] Claudia Campolo, Yevgeni Koucheryavy, Antonella Molinaro, and Alexey Vinel. Characterizing broadcast packet losses in IEEE 802.11 p/WAVE vehicular networks. In *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 735–739. IEEE, 2011.

[19] SY Chen. Kalman filter for robot vision: a survey. *IEEE Transactions on Industrial Electronics*, 59(11):4409–4420, 2011.

[20] Wai Chen and Shengwei Cai. Ad hoc peer-to-peer network architecture for vehicle safety communications. *IEEE Communications magazine*, 43(4):100–107, 2005.

[21] C-F Chiasserini, Rossano Gaeta, Michele Garetto, Marco Gribaudo, and Matteo Sereno. Efficient broadcasting of safety messages in multihop vehicular networks. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, pages 8–pp. IEEE, 2006.

[22] CF Chiasserini, E Fasolo, R Furiato, R Gaeta, M Garetto, M Gribaudo, M Sereno, and A Zanella. Smart broadcast of warning messages in vehicular ad hoc networks. In *Workshop Interno Progetto NEWCOM (NoE)*, 2005.

[23] L Chisalita and Nahid Shahmehri. A peer-to-peer approach to vehicular communication for the support of traffic safety applications. In *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*, pages 336–341. IEEE, 2002.

[24] Junsung Choi, Vuk Marojevic, Carl B Dietrich, Jeffrey H Reed, and Seungyoung Ahn. Survey of spectrum regulation for intelligent transportation systems. *IEEE Access*, 8:140145–140160, 2020.

[25] Federal Communications Commission. Unlicensed national information infrastructure (U-NII) devices in the 5 GHz band, ET Docket No. 13-49, FCC 16-68. *FCC, Oct*, 2018.

[26] Federal Communications Commission et al. Phase I Testing of Prototype U-NII-4 Devices, Report: TR 17-1006. *FCC, Jun*, 2016.

[27] Avik Dayal, Vijay K Shah, Biplav Choudhury, Vuk Marojevic, Carl Dietrich, and Jeffrey H Reed. Adaptive semi-persistent scheduling for enhanced on-road safety in decentralized v2x networks. In *2021 IFIP Networking Conference (IFIP Networking)*, pages 1–9. IEEE, 2021.

[28] ETSI EN. 302 637-2 v1.4.1 intelligent transport systems (ITS); Vehicular Communications; basic set of applications; part 2: Specification of cooperative awareness basic service. 2019.

[29] TS ETSI. 102 687 v1. 1.1: Intelligent transport systems (its); decentralized congestion control mechanisms for intelligent transport systems operating in the 5 ghz range; access layer part. *Access layer part*, 2011.

[30] Randall L Eubank. *A Kalman filter primer*. CRC Press, 2005.

[31] Elena Fasolo, Andrea Zanella, and Michele Zorzi. An effective broadcast scheme for alert message propagation in vehicular ad hoc networks. In *Communications, 2006. ICC'06. IEEE International Conference on*, volume 9, pages 3960–3965. IEEE, 2006.

[32] Peppino Fazio, Mauro Tropea, and Floriano De Rango. A novel PER degradation model for VANETs. *IEEE Communications Letters*, 19(5):851–854, 2015.

[33] Peppino Fazio, Mauro Tropea, Cesare Sottile, and Andrea Lupia. Vehicular networking and channel modeling: a new Markovian approach. In *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pages 702–707. IEEE, 2015.

[34] Huifang Feng, Chunfeng Liu, Yantai Shu, and Oliver WW Yang. Location prediction of vehicles in VANETs using a kalman filter. *Wireless personal communications*, 80(2):543–559, 2015.

[35] Pedro Fernandes and Urbano Nunes. Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):91–106, 2012.

[36] Ryusuke Fukui, Hiroyuki Koike, and Hiromi Okada. Dynamic integrated transmission control (ditrac) over inter-vehicle communications in ITS. In *Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th*, volume 1, pages 483–487. IEEE, 2002.

[37] Mario H Castañeda Garcia, Alejandro Molina-Galan, Mate Boban, Javier Gozalvez, Baldomero Coll-Perales, Taylan Şahin, and Apostolos Kousaridas. A tutorial on 5G NR V2X communications. *arXiv preprint arXiv:2102.04538*, 2021.

[38] Mario Gerla and Jack Tzu-Chieh Tsai. Multicluster, mobile, multimedia radio network. *Wireless networks*, 1(3):255–265, 1995.

[39] Fuad A Ghaleb, Anazida Zainal, Murad A Rassam, and Ajith Abraham. Improved vehicle positioning algorithm using enhanced innovation-based adaptive kalman filter. *Pervasive and Mobile Computing*, 40:139–155, 2017.

[40] Yasaman Ghasempour, Claudio RCM da Silva, Carlos Cordeiro, and Edward W Knightly. IEEE 802.11 ay: Next-generation 60 GHz communication for 100 Gb/s Wi-Fi. *IEEE Communications Magazine*, 55(12):186–192, 2017.

[41] Valentina Graci, Ethan Douglas, Thomas Seacrist, Jason Kerrigan, Julie Mansfield, John Bolte, Rini Sherony, Jason Hallman, and Kristy B Arbogast. Effect of automated versus manual emergency braking on rear seat adult and pediatric occupant precrash motion. *Traffic injury prevention*, 20(sup1):S106–S111, 2019.

[42] IEEE 1609 Working Group et al. IEEE Standard for Wireless Access in Vehicular Environments (WAVE)-Multi-Channel Operation. *IEEE Std*, pages 1609–4, 2016.

[43] Xu Guan, Raja Sengupta, Hariharan Krishnan, and Fan Bai. A feedback-based power control algorithm design for VANET. In *2007 Mobile Networking for Vehicular Environments*, pages 67–72. IEEE, 2007.

[44] Samantha H Haus, Rini Sherony, and Hampton C Gabler. Estimated benefit of automated emergency braking systems for vehicle–pedestrian crashes in the united states. *Traffic injury prevention*, 20(sup1):S171–S176, 2019.

[45] Julian Heinovski, Florian Klingler, Falko Dressler, and Christoph Sommer. A simulative analysis of the performance of IEEE 802.11p and ARIB STD-T109. *Computer Communications*, 122:84–92, 2018.

[46] Gavin Holland, Nitin Vaidya, and Paramvir Bahl. A rate-adaptive MAC protocol for multi-hop wireless networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 236–251. ACM, 2001.

[47] Jinling Hu, Shanzhi Chen, Li Zhao, Yuanyuan Li, Jiayi Fang, Baozhu Li, and Yan Shi. Link level performance comparison between LTE V2X and DSRC. *Journal of Communications and Information Networks*, 2(2):101–112, 2017.

[48] Ching-Ling Huang, Yaser P Fallah, Raja Sengupta, and Hariharan Krishnan. Adaptive inter-vehicle communication control for cooperative safety systems. *IEEE network*, 24(1), 2010.

[49] Xavier Huppé, Jean de Lafontaine, Mathieu Beauregard, and François Michaud. Guidance and control of a platoon of vehicles adapted to changing environment conditions. In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme-System Security and Assurance (Cat. No. 03CH37483)*, volume 4, pages 3091–3096. IEEE, 2003.

[50] Steffi Knorn and Richard H. Middleton. Stability of two-dimensional linear systems with singularities on the stability boundary using LMIs. *IEEE Transactions on Automatic Control*, 58(10):2579–2590, 2013.

[51] Nerijus Kudarauskas. Analysis of emergency braking of a vehicle. *Transport*, 22(3):154–159, 2007.

[52] C Lei, EM Van Eenennaam, W Klein Wolterink, G Karagiannis, Geert Heijenk, and J Ploeg. Impact of packet loss on CACC string stability performance. In *2011 11th International Conference on ITS Telecommunications*, pages 381–386. IEEE, 2011.

[53] Wan Lei, Anthony CK Soong, J Liu, Y Wu, Brian Classon, Weimin Xiao, and T Saboorian. *5G System Design: An End to End Perspective*. Springer, 2021.

[54] Ye Li, Hao Wang, Wei Wang, Lu Xing, Shanwen Liu, and Xueyan Wei. Evaluation of the impacts of cooperative adaptive cruise control on reducing rear-end collision risks on freeways. *Accident Analysis & Prevention*, 98:87–95, 2017.

[55] Chi-Ying Liang and Huei Peng. String stability analysis of adaptive cruise controlled vehicles. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 43(3):671–677, 2000.

[56] Yutong Liu, Kai Shi, Guangping Xu, Sheng Lin, and Shuangxi Li. Analysis of packet loss characteristics in VANETs. In *2018 8th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pages 219–222. IEEE, 2018.

[57] Nikita Lyamin, Alexey Vinel, Magnus Jonsson, and Jonathan Loo. Real-time detection of denial-of-service attacks in IEEE 802.11p vehicular networks. *IEEE Communications letters*, 18(1):110–113, 2013.

[58] Satoshi Matsuda, Hiroyuki Koike, and Hiromi Okada. Vehicular information broadcasting relay (VIBROR) protocol for inter-vehicle-communications. In *Vehicular Technology Conference, 2000. IEEE-VTS Fall VTC 2000. 52nd*, volume 4, pages 2005–2010. IEEE, 2000.

[59] K. Mehrotra and P.R. Mahapatra. A jerk model for tracking highly maneuvering targets. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4):1094–1105, 1997.

[60] Yvonne Mertens, Matthias Wellens, and Petri Mahonen. Simulation-based performance evaluation of enhanced broadcast schemes for IEEE 802.11-based vehicular networks. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 3042–3046. IEEE, 2008.

[61] Rafael Molina-Masegosa and Javier Gozalvez. LTE-V for sidelink 5G V2X vehicular communications: A new 5G technology for short-range vehicle-to-everything communications. *IEEE Vehicular Technology Magazine*, 12(4):30–39, 2017.

[62] Rola Naja et al. *Wireless vehicular networks for car collision avoidance*, volume 2013. Springer, 2013.

[63] Gerrit Naus, René Vugts, Jeroen Ploeg, René van de Molengraft, and Maarten Steinbuch. Cooperative adaptive cruise control, design and experiments. In *Proceedings of the 2010 American control conference*, pages 6145–6150. IEEE, 2010.

[64] NHTSA. LIDAR speed-measuring device performance specifications. Technical Report No. DOT HS 809 811, Washington, DC: National Highway Traffic Safety Administration, Mar. 2013.

[65] Association of Radio Industries and Businesses. 700 MHz intelligent transport systems. ARIB Standard ARIB STD-T109 Version 1.3, ARIB, July 2017.

[66] Hitesh A Patel and Darshak G Thakore. Moving object tracking using kalman filter. *International Journal of Computer Science and Mobile Computing*, 2(4):326–332, 2013.

[67] Lloyd Peppard. String stability of relative-motion PID vehicle control systems. *IEEE Transactions on Automatic Control*, 19(5):579–581, 1974.

[68] Hassan Peyravi. Medium access control protocols performance in satellite communications. *IEEE Communications Magazine*, 37(3):62–71, 1999.

[69] Jeroen Ploeg, Bart TM Scheepers, Ellen Van Nunen, Nathan Van de Wouw, and Henk Nijmeijer. Design and experimental evaluation of cooperative adaptive cruise control. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 260–265. IEEE, 2011.

[70] Jeroen Ploeg, Elham Semsar-Kazerooni, Guido Lijster, Nathan van de Wouw, and Henk Nijmeijer. Graceful degradation of cooperative adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems*, 16(1):488–497, 2014.

[71] Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. Lp string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, 22(2):786–793, 2013.

[72] Lisandro José Puglisi. On the velocity and acceleration estimation from discrete time-position signal of linear encoders. *Journal of Control Engineering and Applied Informatics*, 17(3):30–40, 2015.

[73] Anjan Rayamajhi, Zoleikha Abdollahi Biron, Roberto Merco, Pierluigi Pisu, James M Westall, and Jim Martin. The impact of dedicated short range communication on cooperative adaptive cruise control. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2018.

[74] Anjan Narsingh Rayamajhi. *Exploring Smart Infrastructure Concepts to Improve the Reliability and Functionality of Safety Oriented Connected Vehicle Applications.* PhD thesis, Clemson University, 2019.

[75] M Elena Renda, Giovanni Resta, Paolo Santi, Francesca Martelli, and Alessandro Franchini. IEEE 802.11p VANETs: Experimental evaluation of packet inter-reception time. *Computer Communications*, 75:26–38, 2016.

[76] H-J Reumerman, Marco Roggero, and Marco Ruffini. The application-based clustering concept and requirements for intervehicle networks. *IEEE Communications Magazine*, 43(4):108–113, 2005.

[77] Shahram Rezaei, Raja Sengupta, and Hariharan Krishnan. Reducing the communication required by DSRC-based vehicle safety systems. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pages 361–366. IEEE, 2007.

[78] Steven E Shladover. The California PATH program of IVHS research and its approach to vehicle-highway automation. In *Proceedings of the Intelligent Vehicles92 Symposium*, pages 347–352. IEEE, 1992.

[79] Razvan Stanica, Emmanuel Chaput, and Andre-Luc Beylot. Physical carrier sense in vehicular ad-hoc networks. In *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, pages 580–589. IEEE, 2011.

[80] Razvan Stanica, Emmanuel Chaput, and André-Luc Beylot. Why VANET beaconing is more than simple broadcast. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. IEEE, 2011.

[81] D. Swaroop and J. K. Hedrick. Constant spacing strategies for platooning in automated highway systems. *Journal of Dynamic Systems Measurement and Control-transactions of The Asme*, 121(3):462–470, 1999.

[82] Alireza Talebpour, Dominique Lord, Michael Manser, Sahar Ghanipoor Machiani, Safety through Disruption, et al. Preventing crashes in mixed traffic with automated and human-driven vehicles. Technical report, Safety through Disruption (Safe-D) University Transportation Center (UTC), 2020.

[83] Alberto Lopez Toledo and Xiaodong Wang. Robust detection of MAC layer denial-of-service attacks in CSMA/CA wireless networks. *IEEE Transactions on Information Forensics and Security*, 3(3):347–358, 2008.

[84] ETSI TS. 102 687 v1.2.1: Intelligent transport systems (ITS); decentralized congestion control mechanisms for intelligent transport systems operating in the 5 GHz range. *Access layer part*, 2018.

[85] Yu-Chee Tseng, Sze-Yao Ni, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. *Wireless networks*, 8(2/3):153–167, 2002.

[86] R Vugts. String-stable CACC design and experimental validation. *Diss. Technische Universiteit Eindhoven*, 2010.

[87] Ziran Wang, Guoyuan Wu, and Matthew J Barth. Developing a distributed consensus-based cooperative adaptive cruise control system for heterogeneous vehicles with predecessor following topology. *Journal of Advanced Transportation*, 2017, 2017.

[88] Starsky HY Wong, Hao Yang, Songwu Lu, and Vaduvur Bharghavan. Robust rate adaptation for 802.11 wireless networks. In *Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 146–157. ACM, 2006.

[89] Chaoxian Wu, Yuan Lin, and Azim Eskandarian. Cooperative adaptive cruise control with adaptive kalman filter subject to temporary communication loss. *IEEE Access*, 7:93558–93568, 2019.

[90] Bo Xu, Aris Ouksel, and Ouri Wolfson. Opportunistic resource exchange in inter-vehicle ad-hoc networks. In *Mobile Data Management, 2004. Proceedings. 2004 IEEE International Conference on*, pages 4–12. IEEE, 2004.

[91] Badreddine Yacine Yacheur, Ahmed Toufik, and Mohamed Mosbah. Analysis and Comparison of IEEE 802.11p and IEEE 802.11bd. In *International Workshop on Communication Technologies for Vehicles. Nets4Cars/Nets4Trains/Nets4Aircraft 2020. Lecture Notes in Computer Science*, pages 55–65. Springer, 2020.

[92] Diana Yanakiev, Jennifer Eyre, and Ioannis Kanellakopoulos. Analysis, design, and evaluation of ACVS for heavy-duty vehicles with actuator delays. Research Report UCB-ITS-PRR-98-18, UC Berkeley: California Partners for Advanced Transportation Technology, Berkeley, California, 1998.

[93] Jijun Yin, Tamer ElBatt, Gavin Yeung, Bo Ryu, Stephen Habermas, Hariharan Krishnan, and Timothy Talty. Performance evaluation of safety applications over DSRC vehicular ad hoc networks. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 1–9. ACM, 2004.

[94] Jing Zhao and Guohong Cao. VADD: Vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE transactions on vehicular technology*, 57(3):1910–1922, 2008.

[95] Jing Zhao, Yang Zhang, and Guohong Cao. Data pouring and buffering on the road: A new data dissemination paradigm for vehicular ad hoc networks. *IEEE transactions on vehicular technology*, 56(6):3266–3277, 2007.