Poboljšanje toka prometa i smanjenje broja nesreća komunikacijom među vozilima

Belušić, Marko

Master's thesis / Diplomski rad

2023

UNIVERSITY OF ZAGREB
FACULTY OF ORGANIZATION AND INFORMATICS
VARAŽDIN

Marko Belušić

VEHICLE-TO-VEHICLE COMMUNICATION IN IMPROVING ROAD CONGESTION AND LOWERING NUMBER OF ACCIDENTS

MASTER'S THESIS

Varaždin, 2023

UNIVERSITY OF ZAGREB

FACULTY OF ORGANIZATION AND INFORMATICS

VARAŽDIN

Marko Belušić

Student ID: 0016136348

Programme: Databases and knowledge bases

VEHICLE-TO-VEHICLE COMMUNICATION IN IMPROVING ROAD CONGESTION AND LOWERING NUMBER OF ACCIDENTS

MASTER'S THESIS

Mentor:

Bogdan Okreša Đurić, PhD

Varaždin, September 2023

Marko Belušić

Statement of Authenticity

Hereby I state that this document, my Master's Thesis, is authentic, authored by me, and that, for the purposes of writing it, I have not used any sources other than those stated in this thesis. Ethically adequate and acceptable methods and techniques were used while preparing and writing this thesis.

The author acknowledges the above by accepting the statement in FOI Radovi online system.

Abstract

This paper offers a comprehensive study on Vehicle-to-Vehicle (V2V) communication, a key aspect of Intelligent Transportation Systems. The paper delves into the definition, uses, global state, and technicalities of V2V communication, discussing its limitations and potential solutions. It further explores the practical application of this technology through the implementation of traffic simulations, providing insights into its potential in real-world scenarios. This examination aims to serve as a valuable resource for those interested in the impact and future of V2V communication.

Keywords: v2v; v2x; agents; traffic simulation; python; Carla simulator

Table of Contents

1. Introduction

Vehicle-to-Vehicle (V2V) communication represents an important component of Intelligent Transportation Systems, holding the potential to redefine road safety and traffic efficiency. This technology, coupled with Vehicle-to-Everything (V2X) and Vehicle-to-Infrastructure (V2I) communication, indicates the arrival of a new era in transportation. In this setting, vehicles can interact with each other and the surrounding environment to make data-driven decisions, enhancing the capabilities of human operators.

This paper thoroughly explores V2V communication, outlining its definition, various uses, and the current global landscape. The focus extends into the technical aspects of this innovation, examining methods for information transmission and reception, the associated security implications, and the standard protocols governing message propagation and formatting.

Furthermore, the challenges impeding the widespread adoption of V2V communication are discussed, offering insight into its limitations and suggesting potential remedies.

The practical application of V2V communication is also considered, with detailed walkthroughs of traffic simulations employing this technology. A few simulations are constructed using tools such as the Carla simulator and the SPADE library, including crash avoidance, lane change assistance, and traffic congestion alleviation. These simulations serve to illustrate the potential of V2V communication in real-world scenarios.

Through this examination, the paper analyzes V2V communication, matching theoretical understanding with practical applications. It aims to offer a valuable resource for anyone interested in this transformative technology's potential impact and future direction.

2. Methods and Techniques Used

The exploration and evaluation of Vehicle-to-Vehicle (V2V) communication in this paper involve the application of diverse research methodologies and experimental techniques.

2.1. Literature Review

A significant part of this study involves an extensive literature review. The aim is to cover both the contemporary and historical landscapes of V2V communication. The references include a wide array of academic articles, whitepapers, market reports, and online publications. These sources play a crucial role in explaining and defining complex concepts related to V2V communication, its use cases, global market trends, and the underlying technology.

2.2. Comparative Analysis

When addressing the current state of V2V communication, a comparative analysis of varying V2V technologies, such as Dedicated Short-Range Communications (DSRC) and Cellular V2X (C-V2X), is conducted. This analysis also covers the key players in the market, their unique contributions, and their positions concerning the different technologies. The process entails studying industry reports, press releases, and performance results associated with these technologies and manufacturers.

2.3. Technical Investigation

The technical components of V2V communication require an in-depth examination of specific technical manuals, standards, and security protocols. This part of the research explores the methods for transmitting and receiving information, security implications, and the standard protocols that guide message propagation and formatting. This process involves reading and understanding complex technical documents and connecting these technicalities to their practical implications in real-world scenarios.

2.4. Simulation Setup and Execution

To transition the theoretical knowledge into a practical setting, certain software tools are employed to create traffic simulations using V2V communication. The Carla simulator is chosen due to its broad capabilities for generating realistic traffic scenarios and its compatibility with Python, a commonly used programming language in data science and machine learning.

The SPADE library, known for its agent-based modeling capabilities, is used to simulate the decision-making processes of vehicles. Agent-based modeling allows for the simulation of complex interactions between autonomous agents (vehicles, in this case) and their environment.

3. Vehicle-to-Vehicle Communication

3.1. Definition of Vehicle-to-Vehicle Communication

Vehicle-to-Vehicle (V2V) communication is perceived as a groundbreaking advancement in automotive technology, heralding a new era where road safety is augmented through interconnected vehicles. The capability to identify and react to potential collisions in advance not only has the potential to minimize accidents but also promotes a more streamlined and intelligent transportation ecosystem. Visualization of the concept can be seen in Figure 1.

Figure 1: Visualization of v2v communication; Source: [1]

This is a more detailed description of V2V explained using [2]. Vehicle-to-Vehicle (V2V) communication is an innovative technology that facilitates the wireless transfer of data, such as speed, location, and direction between vehicles. This system allows vehicles to send and receive messages in all directions multiple times per second, establishing a comprehensive awareness of nearby vehicles. With the right software or safety applications, vehicles can interpret these messages to recognize potential collision risks in real time. The system can then trigger various alerts - visual, tactile, or auditory - to warn drivers, enabling them to take preventive measures to avoid accidents. V2V communication messages can cover a distance of more than 300 meters, and they can identify threats hidden by obstacles, landscapes, or weather. This technology complements and enhances existing crash avoidance systems that rely on radars and cameras.

3.2. Accident Prevention Capabilities of V2V

The ability of V2V communication to avert a substantial number of accidents each year underscores the influence of technology in modern transportation. Widespread implementation of V2V could signify a pivotal moment in road safety, transitioning the emphasis from mere survival in accidents to comprehensive prevention, thus preserving numerous lives and fostering a more secure driving environment for everyone.

V2V communication technology has the potential to dramatically improve road safety and save lives. It is estimated that the application of V2V technology could prevent up to 615,000 motor vehicle accidents in the US. In 2019 alone, approximately 6.8 million crashes were reported to the police in the US; these led to 36,096 fatalities and an estimated 2.7 million injuries. By equipping vehicles with connected technologies like V2V, drivers can be better prepared to anticipate potential accidents, significantly reducing the annual loss of life. Importantly, this technology doesn't just help drivers survive accidents, it helps them prevent accidents from occurring in the first place. Moreover, the data shared through V2V communication doesn't reveal the identity of the driver or the vehicle, and there are technical safeguards in place to prevent vehicle tracking and system tampering.[2]

3.3. V2V, V2I and V2X

Vehicle-to-Vehicle, or V2V, is a technology that facilitates the sharing of information between vehicles [3]. Utilizing dedicated short-range communications (DSRC), vehicles can exchange data about their speed and position within a 300-meter radius. This information helps alert drivers to potential hazards, such as impending collisions or adverse weather conditions, contributing to a reduction in accidents and enhancing overall road safety.

Vehicle-to-Infrastructure (V2I) is a system that gathers and transmits data like traffic flow, weather warnings, and the status of traffic signals to drivers. Part of the U.S. Intelligent Transportation Systems (ITS) initiative, V2I helps provide a clearer understanding of traffic conditions and can lead to the development of advanced driver-assistance systems. Future applications of V2I may include innovations like smart parking and more efficient city traffic planning. [3]

Vehicle-to-Everything, or V2X, is a broader concept that encompasses both V2V and V2I technologies. It empowers vehicles to "talk" to all elements of the traffic system, including other vehicles and various infrastructure components. V2X can alert drivers to nearby accidents, hazardous weather, and traffic jams. It also simplifies tasks like toll and parking payments. Though V2X is seen as a key part of the future of autonomous driving, it is still in the developmental stages [3]. In Figure 2 it is possible to see a visualization of V2X communication.

Differences Between V2V, V2I, and V2X are mentioned below in a few bullet points:

- V2V (Vehicle-to-Vehicle): Concentrates on the exchange of information between vehicles, relaying details about speed, location, and potential road hazards.
- V2I (Vehicle-to-Infrastructure): Focuses on the interaction between vehicles and infrastructure elements such as traffic signals, providing insights into traffic patterns and safety considerations.
- V2X (Vehicle-to-Everything): An all-inclusive approach that integrates both V2V and V2I, aiming to create a seamless and intelligent transportation network, including automation of certain payments.

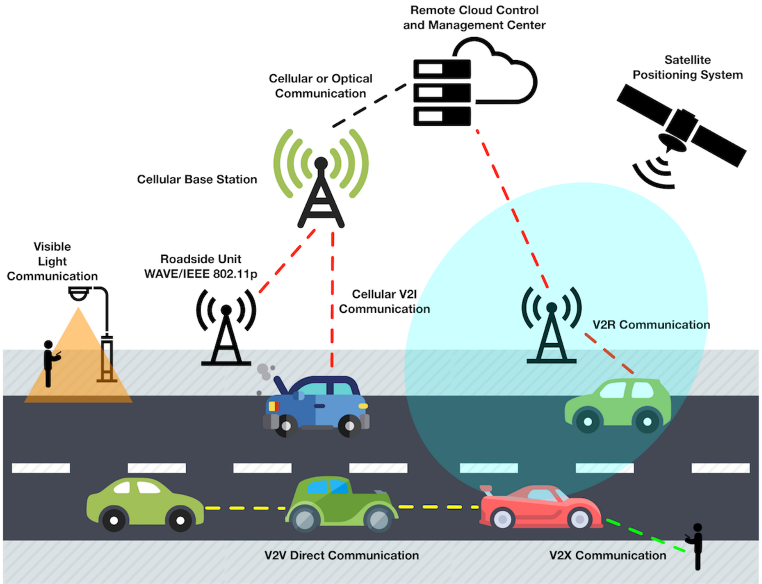


Figure 2: Visualization of V2X Communication; Source: [4]

3.4. Use Cases of V2V Communication and Safety Applications

In this section, there is an overview of the use cases and safety applications of V2V communication.

3.4.1. Use Cases of V2V Communication

The following are high-level use cases for V2V communication [1].

Accident Prevention: V2V communication can detect obstacles or any other medium through a combination of different types of alerts, such as audio, video, or tactile. It warns the driver of any potential threats or even takes action to prevent them. This technology can help two vehicles communicate with each other, reducing the chance of a crash or traffic congestion. In Figure 3 it is possible to see visualization for left turn warning.

Traffic Management: By providing real-time information about traffic conditions, V2V communication helps in traffic management. Law enforcement officers can monitor traffic using real-time data streams and manage traffic effectively. This system also helps drivers maintain distance from other vehicles and reduce traffic jams.

Fuel Consumption: V2V communication can help optimize fuel consumption, especially in truck platoons. The leading truck shows the route that enables the other trucks to consume less fuel to reach the destination. The truck platoon can be seen in Figure 4.

Integration with Various Vehicles: All kinds of vehicles, including trucks, cars, motorcycles, and even bicycles, can use V2V communication. This enhances performance and driver security.

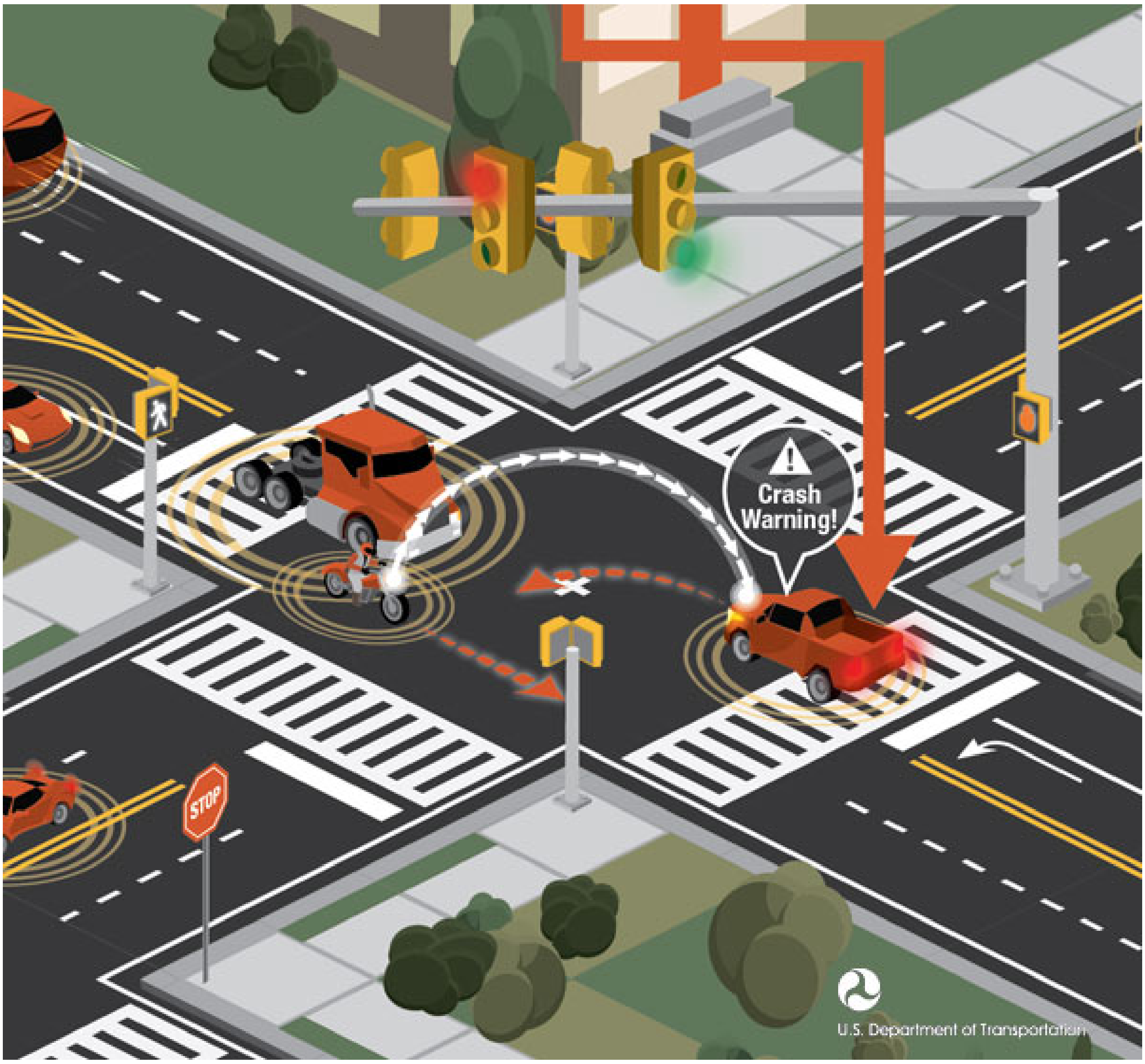


Figure 3: Left-Turn Warning Visualization; Source: [5]

3.4.2. Safety Applications of V2V Communication

The following safety aspects are commonly attributed to V2V communication [1].

Warning Systems: V2V communication can provide various warnings such as red light violation warnings, speed zone warnings, spot weather information warnings, stop sign gap warnings, curve speed warnings, and more. These warnings ensure the safety of the driver and the people inside the vehicle.

Automatic Collision Avoidance: The technology uses radars and cameras to detect and avoid collisions of vehicles. It not only helps drivers and automatic vehicles survive dangerous threats but also avoids creating crashes.

Performance Metrics: Various performance metric tools are present to measure the performance of V2V communication systems. These metrics store information about the type of crash, weather conditions, location, road conditions, and others. These data are used in

Figure 4: Visualization of Truck Platooning; Source: [6]

researching crashes in cities and helps improve transportation facilities to reduce deaths.

Enhanced Security: V2V communication ensures the confidentiality of the driver's information while sending and receiving warning messages. The protocol system increases the performance of the security system of vehicles and saves lives.

3.4.3. Applications of V2V Communication

These applications are similar to the use cases mentioned above, but they are more specific from a different author.

Intelligent Traffic Management Strategies: V2V communication can contribute to solving traffic congestion by enabling properly planned road networks and intelligent traffic management strategies. As stated in the [7], "Traffic congestion has become a problem nowadays, leading significant cities to manipulate the technology to provide better, quicker and more efficient methods to gain access. The main cause of road congestion is the drastic growth of the traffic population. However, it can be solved through proper planned road networks and intelligent traffic management strategies". Real-time road and traffic data sharing among vehicles can lead to more efficient access and reduced congestion.

In-Vehicle Safety Systems: Adopting Intelligent Transportation Systems (ITS), GPS, and in-vehicle safety systems has accelerated the transformation towards smart cars. V2V communication is a key component in this evolution, enhancing safety and driving efficiency. [7]

Simulation Testing for Large-Scale Communication: Specific attention to mobility models, such as lane changes, high influence areas, traffic lights, and topographical information, is required for simulation testing of large-scale V2V communication schemes. These models are essential for the development and evaluation of V2V technologies [7].

Automated Vehicles Integration: V2V communication is integral to the development of automated vehicles, including functionalities like lane centering systems, cruise control, braking systems, parking assist systems, and more. This integration leads to higher levels of automation and safety on the roads [7]. In the section "Current state of Vehicle-to-Vehicle communication in the world" BMW testing is explained.

Assistance for Overtaking Maneuvers: V2V communication can assist in overtaking maneuvers by providing real-time information about other vehicles, positions, and movements. This concept is supported by research as noted in [7]. This application can enhance the safety and efficiency of overtaking on highways and other roads.

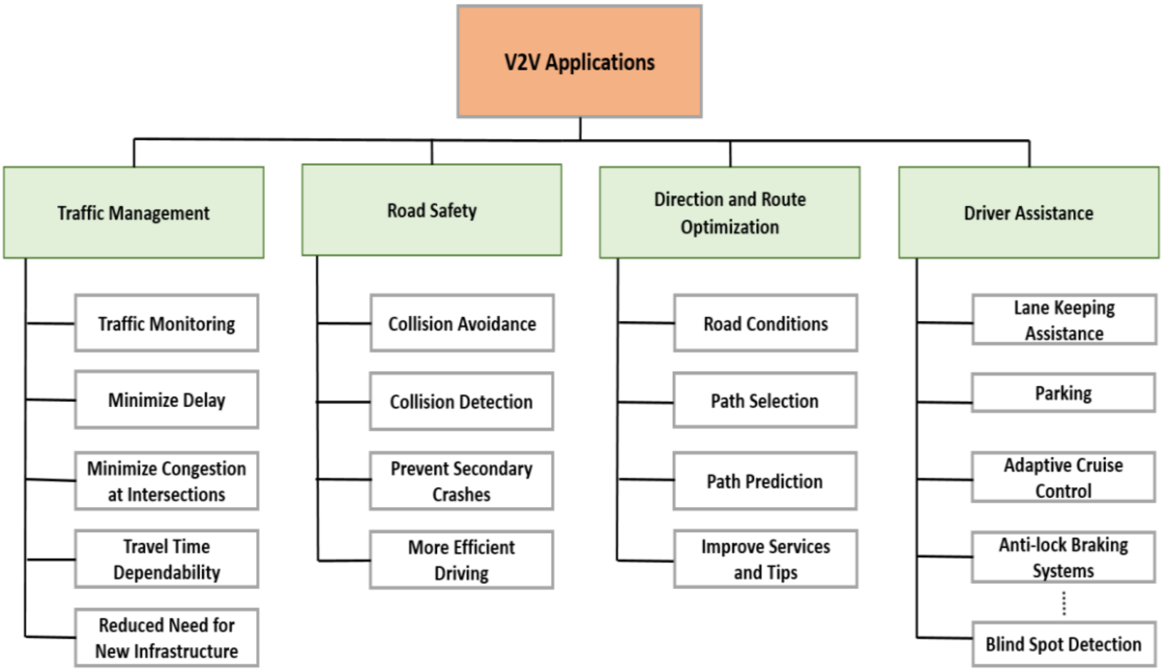


Figure 5: Flow chart diagram of Vehicle-to-Vehicle (V2V) applications; Source: [7]

The chart in Figure 5 is divided into four main categories [7]: Traffic Management, Road Safety, Direction and Route Optimization, and Driver Assistance. Each category has several subcategories, such as Traffic Monitoring, Collision Avoidance, Path Selection, and Blind Spot Detection. The chart is color-coded, with each main category having a different color. The chart is organized in a hierarchical structure, with the main categories at the top and the subcategories below them.

3.5. Current State of Vehicle-to-Vehicle Communication In the World

This section will discuss the current global landscape of Vehicle-to-Vehicle (V2V) communication, comparing the DSRC and C-V2X technologies, evaluating the current state of C-V2X in vehicles, and assessing the global market situation for V2V.

3.5.1. DSRC vs C-V2X

Dynamic Short-Range Communications (DSRC) is a technology based on WiFi communications standards established by the IEEE 802.11 back in 1997 [8]. Tailored enhancements for automotive safety applications were defined in 2009's Wireless Access in Vehicular Environment (WAVE) protocol in the U.S. DSRC was once considered the best way to enable V2X communication, allowing vehicles to connect with one another and with roadside infrastructure without needing a regional network [8].

In 2016, the National Highway Traffic Safety Administration (NHTSA) initiated a move towards mandating all cars sold in 2023 to be equipped with DSRC-based V2V and V2X technology. General Motors was among the first to introduce the concept of V2V communications, leading to the technology's debut in the 2017 Cadillac CTS [8].

On the other hand, Cellular Vehicle-to-Everything (C-V2X) communication began transitioning to 4G and eventually to the Long-Term Evolution (LTE) variant throughout the 2010s. Although initially not as favorable as DSRC, today's 5GLTE is vastly faster and permits direct communication between vehicles and infrastructure even in areas lacking network coverage. In such conditions, C-V2X exceeds the range of DSRC by about 25%.

A significant advantage of C-V2X is that most of the infrastructure required for 5GLTE is installed and maintained by private-sector companies. As stated in [8], "China has gone all-in on C-V2X tech, and the VW group has announced all its autonomy efforts will utilize 5G LTE."

In 1999, the Federal Communications Commission (FCC) reserved 75 MHz of the 5.9 GHz radio spectrum for DSRC-based vehicle-safety usage. However, in November 2020, the FCC freed up 45 MHz of that spectrum for other uses, citing DSRC's lack of adoption. A Joint Waiver Request by several companies and departments led to the FCC's April 24, 2023, order freeing up the remaining spectrum for immediate C-V2X development. [8]

The article [8] directly quotes the situation as, "A recent FCC ruling declares a victor in the DSRC vs. 5G V2X car communication battle." This ruling essentially declared C-V2X the winner in the long-running DSRC-vs-5G V2X communication battle.

With the FCC's ruling, the path is clear for C-V2X to become the standard for vehicle communication. Various Audi's already communicate via cellular-based Car2X technology, and the number of vehicles equipped with C-V2X is expected to ramp up quickly. The technology will enable communication to alert drivers of nearby work- and school-zones, emergency vehicles, and even traffic light timings. [8]

The battle between DSRC and C-V2X has been a significant one in the automotive industry. While DSRC had its merits and early adoption, the advancements in 5G technology and the support from private-sector companies have given C-V2X the edge. The recent FCC ruling has solidified C-V2X's position as the future of vehicle communication, relegating DSRC to history. The transition to C-V2X promises to enhance safety and efficiency on the roads, marking a significant step forward in automotive technology. Now that this is finally resolved development of V2V should progress much faster.

3.5.2. Current State of C-V2X

Currently, about 50%-60% of vehicles in North America are equipped with a cellular modem, with most new vehicles still being shipped with LTE modems [9]. The next couple of years are critical for auto OEM decisions, particularly regarding the equipping of vehicles with 5G and 5.9 GHz modems.

Among U.S. operators, AT&T is the clear market leader, with agreements with major auto OEMs like GM and Ford. Verizon and T-Mobile are also significant players, emphasizing combinations of 5G, MEC, and relationships with cloud providers for low latency C-V2X.

The next phase of C-V2X is expected to begin in the next two to three years, focusing on three main areas [9]:

- V2N Using 5G Networks: Enhanced versions of connected cars will emerge, improving entertainment, information, navigation, and OTA updates. Network slicing could be used to prioritize network performance for automotive applications.
- V2I: This area holds exciting potential for improvements in safety and traffic efficiency, such as traffic signal prioritization. However, the challenge lies in getting municipalities to invest in a scalable way. The author in [9] notes, "China is an interesting bellwether here – and is a good two to three years ahead of the United States."
- Advanced Safety and Automated Driving: While the dream of fully autonomous vehicles may be further off, there will be a steady progression of more automated features. By the end of the decade, dynamic cooperative traffic flow and dynamic intersection management are expected.

The current state of C-V2X is at a pivotal juncture, with significant developments expected in the coming years. The industry is witnessing a shift towards 5G modems. The collaboration between automotive and telecom sectors, along with regulatory support, will be crucial in shaping the future of C-V2X. The words of the author, Mark Lowenstein, encapsulate the situation [9]: "We're behind where we thought we'd be, but the next two years are critical in terms of investment/deployment decisions."

3.5.3. Global Market Situation for V2V

V2V as mentioned above is getting ready to start preventing collisions and managing risks. The global V2V market grew from $20.01 billion in 2022 to $23.19 billion in 2023, with an expected growth to $37.19 billion by 2027. Key market segments include passenger and commercial vehicles, with applications in traffic safety, efficiency, infotainment, and payments. Major players include General Motors, Daimler AG, Toyota, and others.[10]

Recent trends driving the market include growing concern for road safety and technological advancements, such as Nav Wireless Technologies Pvt. Ltd.'s introduction of advanced

V2V communication for armed forces. A notable quote from [10] highlights the impact of global events: "The Russia-Ukraine war disrupted the chances of global economic recovery from the COVID-19 pandemic, at least in the short term, affecting many markets across the globe." This encapsulates the complex interplay of factors influencing the V2V communication market.

Recently V2V market has seen significant growth and competition. As of 2022, Original Equipment Manufacturers (OEMs) actively supporting V2V included major car manufacturers such as Audi, Mercedes-Benz, General Motors, Toyota, Volkswagen, Ford, Nissan, and Honda. Alongside them, technology giants like Qualcomm and Cisco played vital roles in shaping the future of this technology. By 2023, the industry had decided to go with the C-V2X standard, solidifying the direction for V2V communication. Companies investing heavily in C-V2X include Qualcomm, LG Electronics, Huawei, Ericsson, Ford, and General Motors. As the article [11] notes, "OEMs supporting V2V include Audi, Mercedes-Benz, General Motors, Toyota, Volkswagen, Ford, Nissan, and Honda. Technology companies such as Qualcomm and Cisco also are heavily involved." The decision to adopt C-V2X has set a clear path for the future of V2V communication, aligning the efforts of automakers and technology companies.

3.5.4. BMW Testing Left-Turn Assist

BMW Group Research and Technology has been actively working on a left-turn assistant autonomous driver's safety aid to mitigate the risks associated with left turns at complex intersections. The system, tested in a BMW 5 Series sedan, aims to address visibility issues and decision-making challenges by employing sensors and automated braking to enhance safety.

The left-turn assistant activates automatically when the car's sensors detect an intention to turn left. It uses GPS and camera-based image recognition to recognize the lane location, and three front-mounted laser scanners to map the area and identify oncoming vehicles. If a collision risk is detected, the system sounds a warning and applies automated braking. The system can be augmented with vehicle-to-vehicle (V2V) communications, extending the range of vehicle recognition to 820 ft (250 m). As explained by development engineer Udo Rietschel in [12], "The car and the motorcycle communicated with one another via the car-to-x [V2V] interfaces as the motorcycle approached. The car and motorcycle exchanged information on the type of vehicle, its position and speed, as well as dynamic data such as its steering angle and whether the indicators were activated." This V2V communication allows for more precise collision prediction and prevention, including the ability to detect unseen vehicles equipped with V2V. The technology, still in the developmental stage, was publicly demonstrated in Wolfsburg, Germany. This was in 2012. This section shows how V2V can further improve safety.

3.6. Challenges Adopting V2V

Vehicle-to-Vehicle (V2V) communication technology is a system designed to transmit basic safety information between vehicles to facilitate warnings to drivers concerning impending crashes. While the technology has been under research for over a decade, it presents several challenges and limitations that must be addressed before widespread adoption. Here's an overview of the challenges and limitations of V2V technology.

3.6.1. Technical Challenges and Limitations

Unless stated otherwise, the source for mentioned and described limitations in this section is [13].

3.6.1.1. System Limitations

System Limitations: The V2V communication system does have a few limitations to keep in mind. These include the difficulty of installing devices, the need for ongoing updates, and ways to overcome these limitations.

Availability Limitations: The V2V system doesn't work as well if GPS signals are weak or lost, like in cities with tall buildings or areas with thick trees. If the GPS is out for 2 to 5 seconds, the system still works but not as precisely, which can be a problem for apps needing exact lane information.

Congestion Limitations: When there's a lot of cars close together, like in busy freeway traffic, the system could get overwhelmed with too many messages. Making sure the system still works well in these crowded conditions is a challenge.

Operational Range: V2V can work within a 300-meter distance between cars. This is better than in-car sensors that can get affected by bad weather, but the V2V system still has its own issues, like being affected by tall buildings and weak GPS signals.

3.6.1.2. Interoperability of V2V Systems

Interoperability: Making different V2V systems work together is a big challenge. It's important that cars can talk to each other in a way that's fast, reliable, and follows a common standard. For this to work, the messages and the network they use must be the same across different cars and devices. Right now, they use a network standard called IEEE 1609.4 and a wireless layer known as IEEE 802.11p.

Current Maturity Level: Right now, the technology is still in an experimental phase. There have been big tests, but it's not ready to become an official safety standard for all cars yet.

Basic Safety Message (BSM) Maturity: A special kind of message, known as the BSM, lets devices from different makers talk to each other. Early designs for this are in a document called SAE J2735. Future performance rules will be in SAE J2945. However, the existing standard isn't enough on its own to ensure devices work well together, although some cars are already using BSMs to communicate.

Interoperability Performance Requirements: The V2V-Interoperability Report outlines what the equipment inside the car, like the radio and GPS, needs to do to work well together.

Some of these rules have already been included in IEEE and SAE standards. In Ann Arbor, Michigan, over 3,000 vehicles are testing these technologies.

Interoperability Testing Obligation: Automotive manufacturers bear a significant responsibility to rigorously test their V2V devices to ensure seamless communication with devices produced by other companies. Independent testing facilities may be required to impartially evaluate this interoperability. The entity responsible for overseeing system security could necessitate compliance with specific certification criteria for interoperability.

Cooperative System Standards: To facilitate communication between vehicles and additional infrastructural elements such as traffic signals, a distinct set of guidelines known as ITS V2X Cooperative System Standards is essential. These standards serve to ensure that vehicular systems are capable of interpreting and responding to messages from a diverse array of sources.

3.6.1.3. Global Activities and Differences in V2V Systems

Vehicle-to-vehicle (V2V) communication systems are being developed and implemented across the globe, with significant activities in regions like Europe and Asia. Europe is actively working towards implementing V2I (Vehicle-to-Infrastructure), applications centered around mobility by 2015, backed by legislative support and public-private partnerships (details can be found in [13]). In Asia, Japan and Korea stand out as the primary regional frontrunners in terms of development, ultimately paving the way for production implementation. The timeframe has not been reached.

When comparing the United States of America (US) to the European Union (EU), the US mainly concentrates on key V2V applications that can prevent accidents. Initially, the focus was more on V2I (Vehicle-to-Infrastructure) features, but the shift to V2V allows the adoption of Intelligent Transportation System (ITS) safety technologies without the need for expensive infrastructure. On the other hand, the EU prioritizes driver safety tips, supportive messages for drivers (like eco-driving tips), and business-related uses such as insurance. The EU's range of applications is wider and more market-oriented, unlike the US, which is geared towards minimizing crashes. In terms of setting standards, the EU looks at a broader set of applications, whereas the US Department of Transportation (DOT) is mainly concerned with creating safety-focused standards.

When discussing global V2X deployment, the worldwide rollout of V2X (Vehicle-to-Everything) communication differs significantly from the approach taken by the U.S. In the United States, the emphasis is primarily on enhancing safety while keeping costs relatively low. In contrast, the European Union incorporates both mobility applications and strategies that are driven by market demands.

3.6.1.4. Security Options, Limitations, and Design Concept for V2V Communication

When delving into the security options suitable for Vehicle-to-Vehicle (V2V) communication, it becomes evident that robust measures are necessary to facilitate trusted message exchanges crucial for safety. Several options were evaluated, including Symmetric Encryption Systems. While they are notable, they exhibit certain limitations, especially in preserving privacy and managing bandwidth constraints. Group Signature Systems, another potential candidate, was assessed but eventually sidelined. In the end, Asymmetric Public Key Infrastructure Systems (PKI) emerged as the most promising. It was not only adept at preserving privacy but also met the essential criteria of speed and bandwidth constraints. Also, concerns for hacking and bandwidth constraints are mentioned in [14].

Despite the efficacy of PKI systems, current models present some limitations. For starters, they lack the broad applicability necessary to serve as a model for safety-critical applications. Many existing systems are designed for data exchanges among familiar or identifiable parties, usually within the confines of secure or private networks. Tailoring a security model exclusively for V2V poses its own set of challenges. It requires a system that is scalable, extensible to diverse applications, supports operations, maintenance, upgrades, and has mechanisms in place to counteract various risks and threats. Moreover, to be viable, the V2V system must defend against both internal and external threats, support the requirements of an estimated 350 million users, and remain financially sustainable.

The conceptual design for V2V security stands distinct from basic PKI systems. It offers heightened privacy protection, scalability for an enormous user base, and fortified measures to mitigate risks and potential attacks prevalent in today's systems. This design incorporates pioneering technologies such as butterfly keys and linkage values. Additionally, the Security Credentials Management System (SCMS) has been prototyped for the Safety Pilot Model Deployment. This will provide invaluable data to assess the prototype's capabilities in real-world scenarios. At its core, the security strategy for V2V is primarily anchored in PKI. This ensures trusted messaging, feasible operations, and comprehensive privacy protection. It's worth noting that over a decade of research has influenced the design, architecture, and configurations adopted for V2V security. And it's not just internal assessments; the National Institute of Standards and Technology (NIST) has also reviewed and endorsed the PKI approach selected for V2V.

3.6.2. Public Acceptance and Privacy Considerations

Public Acceptance: The success of V2V technology depends on public acceptance. Key aspects of consumer acceptance, potential issues with industry support, and preliminary information on consumer acceptance are explored in the document [13].

Privacy Considerations: Privacy is a significant concern in V2V communication. Important to be careful about the transmission, collection, storage, and sharing of V2V data, privacy policies framework, and NHTSA's interim privacy risk assessment. Privacy and security measures are designed to prevent unauthorized access to information that might enable linkage of data to specific vehicles.

Public Perception and Acceptance: Some respondents in a study (V2V Light Vehicle Driver Acceptance Clinics project conducted from September 2010 to March 2013, more information can be found in [13]) expressed concerns about the unavailability rate of V2V communication, with 43 percent stating that they would accept an unavailability rate of 10 percent or lower. The majority believed that the V2V benefit would not be noticeable until 70-80 percent of vehicles are similarly equipped.

Integration with Vehicle-Resident Systems: V2V technology can be fused with vehicle-resident technologies like sensors, cameras, or radar to provide greater benefits. The longer detection distance and ability to "see" around corners or "through" other vehicles help V2V-equipped vehicles perceive threats sooner. As people are familiar with sensors and cameras already, they could accept V2V faster.

4. Vehicle-to-Vehicle communication from technical perspective

This chapter talks about how vehicles can communicate with each other on the road. It's divided into three parts. The first part looks at the different ways cars can send and receive messages, like using radio signals or mobile networks. The second part talks about keeping these communications safe from hackers and other threats, and what can be done to protect against these risks. The last part goes over the rules and standards for how messages are sent and received, including some testing and information about how messages move from one vehicle to another. Overall, this chapter gives a detailed look at how cars talk to each other, how to keep those conversations secure, and the rules that make sure everything works properly.

4.1. Methods for Transmitting and Receiving Information

The subject of vehicular wireless communication standards is vital for the development of autonomous vehicles (AVs) and intelligent transportation systems (ITS).[15] provided a comprehensive review of various communication standards, including DSRC (Dedicated Short Range Communications), LTE (Long term evolution - C-V2X), and 5G NR-V2X. Below is an exploration of these communication methods, highlighting their features, challenges, and comparisons.

Dedicated Short Range Communications (DSRC) - IEEE 802.11p is a standard for V2X (Vehicle-to-Everything) communication, particularly in the United States. It operates in the 5.9 GHz band and supports low latency communications in high mobility scenarios. DSRC is essential for applications like platooning, traffic management, and emergency services.

Features of DSRC[15]:

- Low Latency: DSRC supports quick communication, vital for high-speed vehicles.
- Investment Ready: Governments and manufacturers have invested significantly in DSRC, making it ready for deployment.

Challenges of DSRC:

- Limited Range: DSRC is suitable for short-range messaging but struggles with high reliability and very low latency in high mobility environments.
- Scalability Issues: DSRC suffers from low data rates, poor scalability, and packet losses, especially in high-density vehicle areas.

A direct quote from the source [15] highlights an issue with DSRC: "The DSRC has many issues. It is appropriate for short-range messaging, however, it does not provide a strong connection, very high reliability, and very low latency, especially in high mobility environments".

Long Term Evolution - LTE (C-V2X) is a cellular technology developed by the 3GPP, facilitating vehicles to function even without cellular infrastructure. It operates at both licensed and unlicensed spectrums.[15]

Features of LTE (C-V2X):

- Network and Direct Communications: LTE uses two types of communications, network communications (using licensed spectrum) and direct communications (using unlicensed spectrum).
- Low-Latency Vehicular Applications: LTE supports low-latency applications through different modes of resource allocation.

Challenges of LTE (C-V2X):

- V2I/I2V Issues: Performance of LTE networks is unclear when network traffic is high, and roaming conditions are complex.
- V2V-Related Issues: Cellular technology faces challenges in handling high data traffic for continuous V2V applications.
- Standardization Issues: Fulfilling all technical and standardization requirements of V2X is difficult for cellular networks.

The author in [15] states, "Existing cellular infrastructure does not support many V2X applications as these applications require low latency in high mobility and/or high congestion scenarios".

IEEE 802.11p is a standard specifically designed for vehicular communications [16]. It's part of the Wireless Access for Vehicular Environments (WAVE) and is primarily used for safety-oriented applications. However, the standard has limitations, especially in providing real-time constraints for safety applications. [16] states, "Despite its limited applicability, several applications based on the IEEE 802.11p standard are on the market, but some projects are testing yet with few vehicles".

4G/5G-D2D represents a trend in vehicular communications that leverages existing and emerging wireless standards, including 4G, 4G+, and 5G technologies. This approach aims to exploit Mobile Network Operators (MNOs) existing telecommunication infrastructures and network data to enhance intelligence on the move, providing novel Advanced Driver Assistance Systems (ADAS) solutions.

The concept of 4G/5G-D2D focuses on the diverse performance requirements of vehicular networking applications and the significant cost of deploying specialized road infrastructure. Research has been moving towards exploiting existing/emerging mobile communication standards, such as LTE—X2 interface and 5G Device-to-Device (D2D), as suitable mechanisms for delivering automotive applications, particularly autonomous driving (AD). [16] states, "It is envisaged that next generation mobile technologies (4G/4G+, 5G), including D2D networking and very low latency communications, will constitute alternative technology solutions to 802.11p".

[16] highlights, "30% increased security, privacy and confidentiality. The solution is protected against jamming and tapping through the utilization of the mobile communication infrastructure mechanisms, compared to current vehicular networking approaches (e.g., IEEE 802.11p)".

With 4G/5G-D2D, drivers, businesses, and public service providers have the possibility to communicate with each other and share useful information. The solution overcomes challenges by introducing a unified frontend to the system. It also enables scenarios like Automated Route Guidance for Emergency Response Vehicles, where real-time traffic maps can be generated with high accuracy. The fundamental novelty of 4G/5G-D2D lies in the utilization of MNOs telecom infrastructures for message transmission, instead of other costly V2I technologies or unreliable V2V technologies. The data sources utilized include a mobile smartphone inside the vehicle, the vehicle itself (via an OBD-II device), and MNO-related data.

4.1.1. Less Popular Methods for Transmitting and Receiving Information

LED-Enabled Visible Light Communications (VLC) is a technology that uses visible or IR LEDs for communication [16]. It can be implemented in various vehicular connections, including downstream connections from traffic lights to vehicles and connections from road and traffic signs to vehicles. VLC offers high reliability, low infrastructure cost, very low carbon emissions, and resilience to interference. [16] explains, "The VLC links can be used for: Downstream connection from the traffic lights to the vehicles using the LEDs of the lights as a means of transmitting data. All three LED colors must be used here (red, green, yellow). Overall, VLC is a valid candidate for complementing current solutions in the world of transportation".

Bluetooth technology is a wireless communication method that is simple, secure, and ubiquitous. It can be found in billions of devices ranging from mobile phones to medical devices and home entertainment products. In the context of vehicular communications, Bluetooth has been implemented for various applications, including hands-free profiles for mobile phones in cars. The development has been coordinated by the Car Working Group (CWG) since 2000, implementing different profiles and features. [16] states, "Bluetooth technology is a wireless communications technology that is simple, secure, and can be found almost everywhere. The key features of Bluetooth technology are ubiquitousness, low power, and low cost". In V2I systems, Bluetooth can be used to provide communication channels between cars and traffic signal systems, and several manufacturers offer Bluetooth-capable traffic control devices.

2G and 3G technologies represent earlier generations of mobile communication infrastructures. They have been used for vehicle-to-back-office communication as part of Wide Area Networking (WAN) technologies. However, these technologies suffer from location accuracy, which could be improved by secondary mechanisms such as GPS. [16] mentions, "Further, Wide Area Networking (WAN) technologies such as 2G/GPRS/EDGE, 3G/UMTS/HSPA/HSPA+, and 4G/LTE have also been used for vehicle to back office communication, but these suffer from location accuracy which could be improved by secondary mechanism such as GPS". Unlike its predecessors, 5G technology enhances location accuracy through features like higher device density, higher frequency, and lower latency, which collectively contribute to more timely and accurate location updates.

4.2. Security in Vehicle-to-Vehicle Communication

With the development of V2V, there are several ways in which the system can be attacked, posing significant security concerns. Here's an overview of the possible ways of attacking the system in V2V, based on [17].

Attacks Targeting Availability:

- Denial of Service (DoS) and Distributed Denial of Service (DDoS): These attacks aim to disrupt the network, making users unable to communicate. The effect of these attacks is high, and they can break down the network entirely.
- Spamming: Spamming consumes the bandwidth of the network, although its impact is considered low.
- Black Hole Attack: This attack makes data lost, creating a first step for a man-in-the-middle attack. Its impact is high.
- Sybil Attack: The Sybil attack provides an illusion of many vehicles to force the vehicles off the road for the attacker's goal. This attack has a high impact.

Attacks Targeting Authentication:

- Node Impersonation: This involves changing the driver's identity and car's identity to deceive authorities, especially when an accident happens. The impact is considered medium.
- Message Suppression and Alteration: These attacks involve changing or modifying existing data to deceive users. The impact is medium.
- Relay Attack: This involves capturing and replaying packets to puzzle authorities and prevent vehicle identity verification in any accident. The impact is high.

Attacks Targeting Confidentiality:

- GPS Spoofing: This attack involves taking control of the identity and location of the vehicle, with a high impact.
- Traffic Analysis: Using packets to analyze important information like vehicle ID and location, this attack has a high impact.
- Social Attack: This attack confuses the victim to make the driver disturbed, with a low impact.

Attacks on In-Car Systems:

- Controller Area Network (CAN): Attackers can exploit CAN vulnerabilities to control various systems in a car, such as electric window lift, warning lights, brake, and airbag control system.
- Tire Pressure Monitoring System (TPMS): Attackers can control TPMS via eavesdropping and spoofing methods, as TPMS lacks cryptographic mechanisms.

Attacks on Vehicle to Infrastructure (V2I) Interface can include breaking down the network, disrupting the network, affecting safety services provided by Road Side Units (RSUs), and stealing sensitive information.[17]

The development of V2V communication technology promises a better future for transportation systems. However, the potential risks of cyber-attacks are increasing, and the security challenges must be addressed. As the source [17] concludes, "Although this technology offers several benefits to users, the security issues of vehicle communication should be considered. Therefore, cyber protection is crucial for the future".

4.2.1. AutoCrypt Solution

AutoCrypt offers a possible solution. AutoCrypt PKI (Public Key Infrastructure) is a specialized technology dedicated to the mobility ecosystem. It plays a vital role in user verification across various connected services, including V2X (Vehicle-to-Everything) communications, Plug&Charge (PnC), and digital keys.

AutoCrypt PKI ensures trust within the V2X and PnC environment. With AutoCrypt PKI's ecosystem, connected participants such as vehicle OBUs (On-Board Units), infrastructure RSUs (Road-Side Units), and pedestrian devices can obtain security certificates from root certificate authorities (root CA). These certificates are attached to their messages as digital signatures, and the root CAs continuously manage them within the ecosystem.[18]

A direct quote from [18] explains the process: "AutoCrypt PKI enables the issuance of both enrolment certificates and authorization certificates to OBUs and RSUs. Enrolment certificates are issued when the units are first registered, while authorization certificates are issued every time a message is exchanged to allow for the system to authenticate each user."

The certificates issued by AutoCrypt PKI can be broken down into different types, including identification certificates, pseudonym certificates, and application certificates. These certificates act as IDs that prove the identity of the participants and the legitimacy of the messages. The infrastructure also continuously manages certificates with an updating list of revoked devices.

As stated in the [18], "Having demonstrated interoperability across international protocols including SCMS, CCMS, and C-SCMS, AutoCrypt PKI is a crucial component for V2X, Plug&Charge (PnC, ISO 15118), and a wide range of vehicular communications, ensuring mutual safety during message exchanges."

4.3. Message Propagation and Format

4.3.1. Message format

4.3.1.1. SAE, SAE J2945/1 and SAE J2735

The Society of Automotive Engineers (SAE) has established a set of standards for vehicle-to-vehicle (V2V) communication, which is crucial for the development and implementation of autonomous vehicles and intelligent transportation systems. These standards are designed to ensure interoperability and a common platform for safety, mobility, and commercial priorities.[19]

The SAE (Society of Automotive Engineers) standard, specifically the SAE J2945/1, is a set of guidelines that outline the minimum functional and performance requirements for transmitting and receiving the Basic Safety Message (BSM) by light vehicles. This standard was published in March 2016 and has been the result of over a decade of research work and real-world testing of V2V (Vehicle to Vehicle) Safety System BSMs. The research involved multiple OEMs, academia, suppliers, and the US Department of Transportation. Ford Motor Company released a presentation with overview of the mentioned standard. [20]

The SAE J2945/1 standard covers various aspects of V2V communication. It includes functional requirements such as standards profiles, positioning and timing, radio characteristics, congestion control, and security and privacy. It also outlines performance requirements, including radio/RF (transmitter and receiver performance requirements) and kinematic data (data accuracy requirements for position, speed, accelerations, yaw rate, etc.).

The SAE J2945/1 standard also specifies the DSRC (Dedicated Short-Range Communications) radio data rate, channel, and access category parameters. It includes guidelines for congestion control and scheduling BSMs for transmission, privacy measures, security (pointing to the IEEE1609.2 standard), and RF performance for both transmitter and receiver.[20]

To ensure full system interoperability at the safety applications level, the SAE J2945/1 standard recommends test procedures to verify compliance with minimum performance requirements. The SAE J2735 (OTA Data Dictionary) enables OTA interoperability, while the SAE J2945/1 enables system and safety applications interoperability.

The SAE J2945/1 standard, with its set of stable requirements, could be used as the foundation for other safety communications development such as B2V (Bicycle to Vehicle), by focusing on specific crash scenarios and use cases. The stakeholders are encouraged to create a proper forum to jointly develop the B2V OTA message and its corresponding set of minimum performance requirements, which would greatly facilitate the adoption and publication through the SAE standards process.[20]

4.3.1.2. Basic Safety Message - BSM

One of the key components of these standards is the Basic Safety Message (BSM), which is used to exchange safety data regarding vehicle state. The BSM is broadcast to surrounding vehicles with data content as required by safety and other applications. The BSM contains critical core data elements deemed to be needed with every BSM issued, often referred to as the "BSM Part One". The BSM includes data such as the vehicle's temporary ID, second mark, latitude, longitude, elevation, and positional accuracy. However, due to the multiplicity of sensor sources, the precise moment in time when each of these values is determined is not likely to be aligned in most V2X devices. Therefore, developers are cautioned not to presume that data such as that found in the BSM reflects a common measurement epoch.[19]

The propagation of these messages is facilitated by the Wireless Access in Vehicular Environments (WAVE) system, which is designed to provide the required interoperable wireless networking services for transportation. The WAVE system supports the high-availability, low-latency communications requirements of vehicle safety applications, such as pre-crash collision mitigation, intersection collision avoidance, and cooperative collision avoidance. The range of this system is generally considered to be distances of less than 1000 meters.[19]

The format of the messages is defined in the SAE standard [19]. A message is a well-structured set of data elements and data frames that can be sent as a unit between devices to convey some semantic meaning in the context of pre-defined applications. The standard also provides several data concepts to provide durations or intervals of time, supporting time ranges from 1 minute to a span of many days. The SAE standard also includes a set of rules for conformance to the format, which requires that required data fields are present, no data field is included that is not either required, explicitly optional, or classified as regional content, data fields appear in the indicated order, and numerical values are within specified ranges, while using the proper units and definitions.[19]

The BSM is structured into two parts [19]: Part I and Part II. Part I, also known as the BSM core data, is sent at all times with each message. It contains the critical core data elements deemed to be needed with every BSM issued. This includes the message count (msgCnt), temporary ID (id), second mark (secMark), latitude (lat), longitude (long), elevation (elev), and positional accuracy (accuracy).

Part II of the BSM is optional and is included as needed according to policies that are beyond the scope of the standard. A BSM without Part II optional content is still considered a valid message. The transmission rates of these messages are typically 10 times per second when congestion control algorithms do not prescribe a reduced rate.

The BSM message format is represented in ASN.1 (Abstract Syntax Notation One), a standard interface description language for defining data structures that can be serialized and deserialized in a cross-platform way. It is widely used in telecommunications and computer networking, and especially in network management.[19]

4.3.2. Message Propagation

Message propagation in V2V communication involves the exchange of vital safety and operational information between vehicles, allowing each vehicle to have detailed knowledge of the surrounding traffic situation [21].

V2V communication can be classified into different propagation mechanisms [21], including periodic broadcast, priority-based broadcast, event-driven broadcast, event-driven geocast, and periodic unitcast to Road Side Units (RSUs) (V2I). The choice of propagation mechanism depends on the specific requirements of the traffic information propagation scenario.

In a typical V2V communication setup, vehicles broadcast their heartbeat messages, allowing each vehicle to identify a list of its neighboring vehicles. By measuring the signal-to-noise ratio through the heartbeat messages received from other vehicles, a vehicle can recognize a set of vehicles to which it can transmit and receive data items.

The propagation of messages in V2V communication is influenced by several factors [21], including vehicle density, traffic workload, and the dwell time of vehicles in the service region. For instance, a higher traffic workload results in a higher vehicle density, leading to more neighbors for each vehicle. This increases the chance of having common requests among neighboring vehicles, thereby enhancing the effectiveness of V2V data dissemination.

The placement of antennas and their beam direction also significantly affect the V2V communication channels. Multiple Input Multiple Output (MIMO) antenna systems [21] offer high data rates and good quality of service in non-line-of-sight environments, making them suitable for V2V communications.

5. Implementation of Traffic Simulation Using V2V Communication

This chapter explores the detailed process of setting up and conducting traffic simulations that utilize Vehicle-to-Vehicle (V2V) communication. It goes into the applications used for development, the logic behind vehicle behaviors, and several key simulation scenarios.

5.1. Applications Used for Development

An overview of the essential software and tools leveraged in creating and running the V2V traffic simulations.

5.1.1. Carla Simulator

The Carla Simulator [22] is an open-source platform specifically crafted to test and validate autonomous driving algorithms. It boasts high-fidelity visuals and offers a spectrum of realistic weather conditions, aiming to emulate real-world dynamic environments. Originating as a tool to bridge the gap between the real world and simulations.

Authors in [23] show the most important use case and explain the architecture. In this thesis, Carla was used to simulate the urban environment and highway environment with realistic car behavior and graphics. It offers an extensive API that helps create needed scenarios and implement agent logic inside cars.

5.1.2. SPADE Library

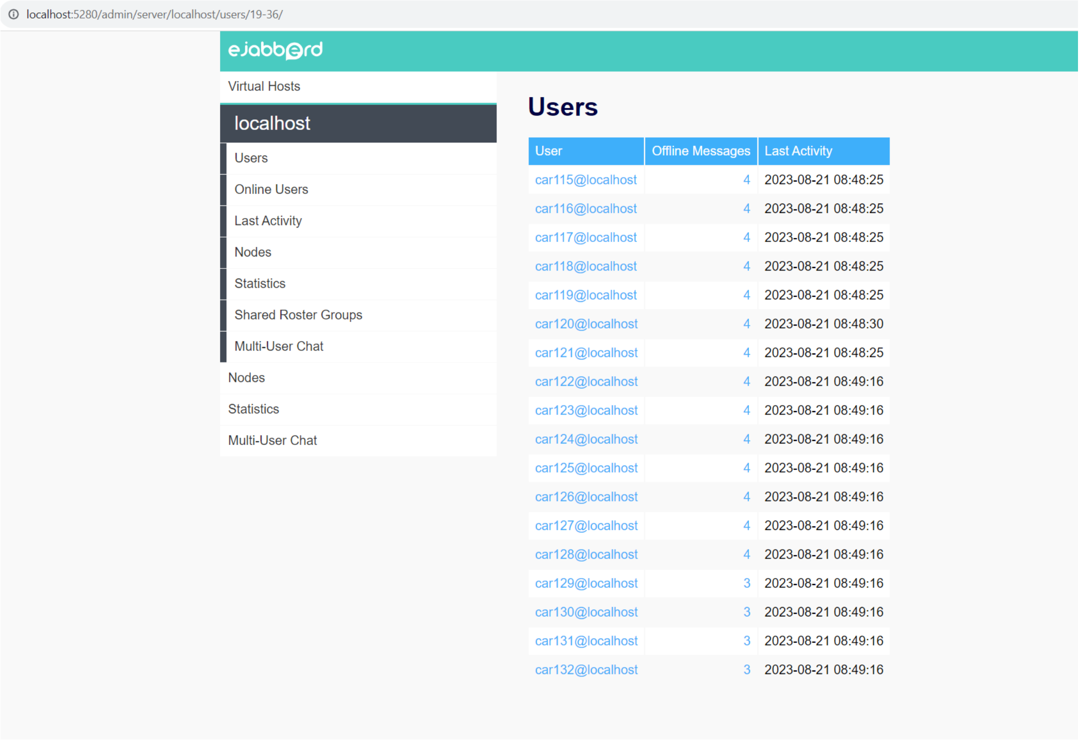
The SPADE Library, or Smart Python multi-Agent Development Environment [24], is a foundational tool for building multi-agent systems. Its core mission is to simplify and streamline the development process for scenarios where numerous entities communicate and make decisions.

For V2V simulations, it provides the framework for defining precise vehicle behaviors, ensuring that V2V messaging and the decision-making processes align closely with real-world expectations. In this thesis, it was used to create a vehicle brain, meaning the logic for parsing, sending, and reacting to BSM messages.

5.1.3. Ejabberd XMPP server

Ejabberd [25] is an open-source Jabber/XMPP server written primarily in the Erlang programming language, designed for high performance and extensibility. It is commonly used to power real-time messaging applications, including instant messaging (IM), voice and video calls, and IoT communication. Configuration and management can be done via a web-based admin interface, command-line tools, or external APIs.

The reason it was chosen over Prosody is out-of-the-box support for Windows via .bat script that will create a Docker container that you run. A screenshot of the web interface can be seen in Figure 6. Using web interface it is possible to see all users, when they were last active, queue of messages and among other things, manage users. There is also an API that can provide more detailed and granular access to specific setups and features.



User	Offline Messages	Last Activity
car115@localhost	4	2023-08-21 08:48:25
car116@localhost	4	2023-08-21 08:48:25
car117@localhost	4	2023-08-21 08:48:25
car118@localhost	4	2023-08-21 08:48:25
car119@localhost	4	2023-08-21 08:48:25
car120@localhost	4	2023-08-21 08:48:30
car121@localhost	4	2023-08-21 08:48:25
car122@localhost	4	2023-08-21 08:49:16
car123@localhost	4	2023-08-21 08:49:16
car124@localhost	4	2023-08-21 08:49:16
car125@localhost	4	2023-08-21 08:49:16
car126@localhost	4	2023-08-21 08:49:16
car127@localhost	4	2023-08-21 08:49:16
car128@localhost	4	2023-08-21 08:49:16
car129@localhost	3	2023-08-21 08:49:16
car130@localhost	3	2023-08-21 08:49:16
car131@localhost	3	2023-08-21 08:49:16
car132@localhost	3	2023-08-21 08:49:16

Figure 6: Ejabberd Web Interface

5.2. Vehicle Brain Setup

Resources from a technical perspective about agents, and behaviors were from [26] and [24].

Multiagent systems are composed of multiple interacting computing elements called agents. These agents possess two primary capabilities [27]. Firstly, they have a degree of autonomy, meaning they can make decisions on their own to achieve their design objectives. Secondly, they can interact with other agents. This interaction isn’t just simple data exchange; agents can engage in activities analogous to human social behaviors, such as cooperation, coordination, and negotiation. The concept of multiagent systems is relatively new, emerging around 1980 and gaining widespread recognition by the mid-1990s. These systems are seen as a promising software paradigm for harnessing the potential of vast open distributed systems like the Internet. Beyond just internet applications, multiagent systems provide a natural metaphor for creating a variety of artificial social systems, making them applicable across various domains. In essence, multiagent systems can be viewed as artificial societies where agents interact in complex environments.

Vehicles in the simulations are set to run on autopilot to replicate real life as much as possible. The autopilot feature is integrated inside Carla. On top of that there is custom logic using SPADE agents. Every vehicle is an agent which sends BSM messages. For every simulation, there will be a separate code so it is easier to understand which logic is necessary for which simulation. On Listing 1 it is possible to see CarAgent class setup. It consists of two behaviors. One behavior is periodic and that behavior will send BSM messages, and the second behavior is cyclic, that behavior is waiting and parsing messages.

Listing 1: Crash prevention CarAgent class

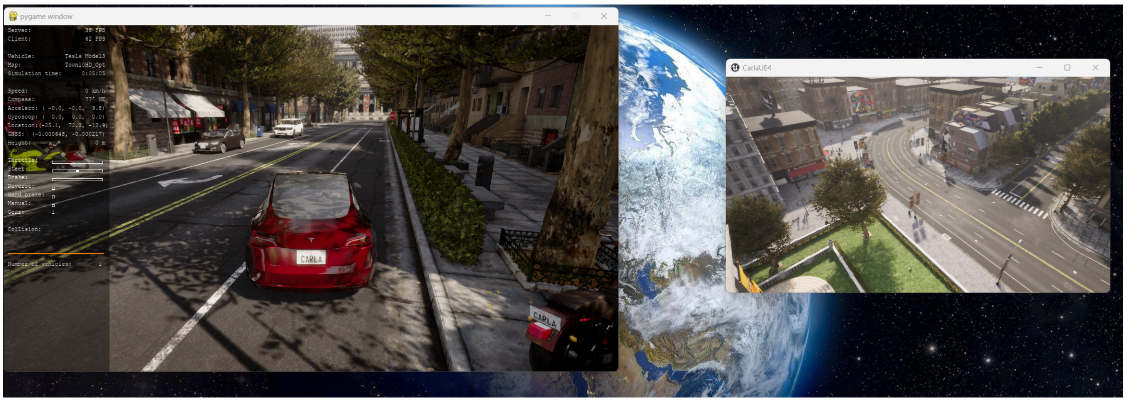
```
bsm_template = spade.template.Template()
bsm_template.metadata = {"performative": "inform", "ontology": "bsm"}

class CarAgent(spade.agent.Agent):
    def __init__(self, jid, password, carla_vehicle):
        super().__init__(jid, password)
        self.carla_vehicle = carla_vehicle
        self.receivers = []

    class SendBSMBehaviour(spade.behaviour.PeriodicBehaviour):
        async def run(self):
            try:
                msg = self.create_bsm_message()
                if msg.body == "":
                    return
            except VehicleDestroyed:
                await self.agent.stop()
                return
            await self.send_message_to_all(msg)

        async def on_end(self):
            pass

        async def on_start(self):
            pass

        def create_bsm_message(self):
            msg = spade.message.Message()
            if not self.agent.carla_vehicle:
                msg.body = ""
            if not self.agent.carla_vehicle.is_alive:
                raise VehicleDestroyed("Please stop agent")
            location = self.agent.carla_vehicle.get_transform().location
            crash = False
            if hasattr(self.agent.carla_vehicle, "crash") and self.agent.
             carla_vehicle.crash:
                crash = True
                self.agent.carla_vehicle.crash = False
            msg.set_metadata("performative", "inform")
            msg.set_metadata("ontology", "bsm")
            msg.body = f"{self.agent.name}*Location:\{location.x\},\{location.y\},\{
             location.z\}*Crash:\{crash\}"
```

```
            return msg

        async def send_message_to_all(self, msg):
            for a in self.agent.receivers:
                msg.to = a
                await self.send(msg)

    class ParseBSM(spade.behaviour.CyclicBehaviour):
        async def run(self):
            msg = await self.receive(timeout=10)
            if msg:
                msg_splitted = msg.body.split("*")
                if msg_splitted[2] == "Crash:True":
                    print("breaking")
                    self.agent.carla_vehicle.set_autopilot(False)
                    control = carla.VehicleControl(throttle=0.0, steer=0.0, brake
                     =1.0, hand_brake=True)
                    self.agent.carla_vehicle.apply_control(control)
                    await asyncio.sleep(5)
                    self.agent.carla_vehicle.set_autopilot(True)
                # print(f"{self.agent.name} got message: {msg.body}")
    async def setup(self):
        self.add_behaviour(self.SendBSMBehaviour(period=0.5), None)
        self.add_behaviour(self.ParseBSM(), bsm_template)
```

5.3. Creating Crash Avoidance Simulation

This simulation consists of 3 main files, which can be found in the Github repository provided at the end of this document. Three files: crash_prevention.py, manual_control.py and crash_prevention_spade.py. crash_prevention_spade.py contains SPADE agent logic and a snippet can be found as Listing 1. Inside manual_control.py there is an example code given by Carla developers, which was modified to accommodate for this simulation. That code gives the ability for a human operator to operate with a vehicle. In this simulation, a human-operated vehicle is the vehicle that caused the accident. The setup for executing this simulation can be seen on Figure 7. On the left, there is a manually controlled vehicle, and on the right is Carla simulator world in spectator mode.

Special care was given in manual_control.py to threading. Because there is a pygame loop running and SPADE agents do not have time to execute and if async is used, the game will of course, start lagging. So SPADE agent is running on a different thread which can be seen on Listing 2. Also, we can notice that thread automatically tries to fetch all other vehicles in a game automatically, if vehicles are removed or added this agent will always have the latest list. With this setup game runs without any performance issues and SPADE agent is available at all times. The try, except, and finally blocks in the code ensure that the SPADE agent thread will be properly joined back into the main thread, even if the program is interrupted, ensuring clean termination of all running processes. SPADE agent is also going to be automatically stopped in this case because when the vehicle is destroyed custom exception VehicleDestroyed is raised



```
Server:                 38 FPS
Client:                 62 FPS

Vehicle:          Tesla Model3
Map:              Town10HD_Opt
Simulation time:       0:05:05

Speed:                  0 km/h
Compass:                77° NE
Accelero: ( -0.0, -0.0,  9.8)
Gyroscop: (  0.0,  0.0,  0.0)
Location:(-25.1,  72.3, -12.9)
GNSS:  (-0.000648, -0.000217)
Height:                    0 m

Throttle:
Steer:
Brake:
Reverse:
Hand brake:
Manual:
Gear:         1

Collision:

Number of vehicles:        1
```

Figure 7: Setup for Crash Avoidance Simulation

that will automatically stop the agent and allow agent to exit and close the connection with XMPP server. The V2V system operates on a message exchange framework that allows for real-time data transmission between connected vehicles.

Listing 2: Manual control Threading

```
    try:
        spade_thread = threading.Thread(target=run_spade)
        spade_thread.start()
        game_loop(args)

    except KeyboardInterrupt:
        print(’\nCancelled by user. Bye!’)
    finally:
        spade_thread.join()
def run_spade():
    asyncio.run(create_spade_agent())
async def create_spade_agent():
    global world
    while not world:
        print("waiting for world")
        await asyncio.sleep(1)
    # create agent
    car_id = world.player.id
    car_obj = world.player
    agent = CarAgent(f"car{car_id}@localhost", f"pass{car_id}", car_obj)
    try:
        await agent.start(auto_register=False)
    except spade.agent.AuthenticationFailure:
        register_user(f"car{car_id}", f"pass{car_id}")
        await agent.start()
    while True:
        receiver_vehicles = get_receiver_vehicles()
        if receiver_vehicles:
            agent.receivers = receiver_vehicles
            print("Loaded vehicles")
        if not agent.is_alive():
            break
        await asyncio.sleep(2)
```

Inside crash_prevention.py other vehicles are created at predefined locations and all set up for autopilot, speed of the cars and SPADE agent creation is handled. On Listing 3 there are 16 predefined locations for vehicles that were manually chosen behind corner to be able to simulate the scenario efficiently. For every created vehicle there is an object from CarAgent class created which is the brain for V2V communication. That process can be seen on Listing 4.

Listing 3: Spawn locations for Crash Prevention Simulation

```
custom_transforms = [carla.Transform(carla.Location(x=19.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=13.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=7.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=1.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-6.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-12.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-18.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-24.0, y=141.3, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=19.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=13.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=7.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=1.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-6.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-12.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                     carla.Transform(carla.Location(x=-18.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0)),
                    carla.Transform(carla.Location(x=-24.0, y=137.4, z=0.6),carla.Rotation(pitch=0.0, yaw=0.0, roll=0.0))]
```

Listing 4: SPADE Agent Creation for Crash Prevention

```
        for response in client.apply_batch_sync(batch, synchronous_master):
            if response.error:
                logging.error(response.error)
            else:
                vehicles_list.append(response.actor_id)
                car = world.get_actor(response.actor_id)
                agent = CarAgent(f"car{response.actor_id}@localhost", f"pass{response.actor_id}", car)
                try:
```

```
                await agent.start(auto_register=False)
            except spade.agent.AuthenticationFailure:
                register_user(f"car{response.actor_id}", f"pass{response.actor_id}")
                await agent.start()
            agents.append(agent)
    for agent in agents:
        receivers = [f"{a.name}@localhost" for a in agents if a != agent]
        agent.receivers = receivers
```

With this setup, the simulation was successful in preventing chain collision. **The link to the video is in the appendices**. In Figures 8, 9 and 10 it is possible to see 3 time-frames without using V2V communication. At the time of the crash, 2 seconds after and 5 seconds after. Chain collisions happened, meaning that multiple cars crashed into each other.

Figure 8: Time of The Crash Without Using V2V Communication

In Figure 9, multiple cars had already crashed.

In Figure 10, a chain collision occurred.

In Figures 11, 12 and 13 it is possible to see 3 time-frames with using V2V communication. At the time of the crash, 2 seconds after and 5 seconds after. Chain collision was prevented and there was only 1 immediate crash after the initial crash.

In Figure 12, the situation is already better than the example without V2V coomunication.

In Figure 13, there is no chain collision. It was prevented and the difference is substantial when compared to a scenario without using V2V communication.

Using V2V communication cars behind the corner got information that the crash happened, and they started breaking in advance, which helped to prevent chain collision.

One crucial metric for evaluating the simulation's success is the drastic reduction in the incidence of chain collisions. This is quantitatively assessed by measuring the number of

Figure 9: 2 Seconds After The Crash Without Using V2V Communication

Figure 10: 5 Seconds After The Crash Without Using V2V Communication

collisions occurring within a specific time frame post the initial crash. In a non-V2V environment, this number was notably higher, illustrating the efficacy of vehicle-to-vehicle communication in mitigating further accidents.

Another parameter to consider is the time taken for the vehicles to come to a complete stop after receiving the emergency brake signal. This ’reaction time’ is crucial in evaluating the system’s efficiency and also in comparing the autonomous response to typical human reaction times in similar situations. Comparison to the human operator was not simulated here, when the crash signal was received car started breaking immediately.

Figure 11: Time of The Crash Using V2V Communication

Figure 12: 2 Seconds After The Crash Using V2V Communication

Figure 13: 5 Seconds After The Crash Using V2V Communication

5.4. Lane Change on Highway Helper Simulation

Using V2V communication location of other vehicles is known and that information can be used to inform drivers of the threat on the highway while trying to change the lane. That scenario was implemented in the following way. Three new files were created: "lane_change_simulation.py", "manual_control_lane_change.py" and "lane_change_simulation_spade.py"

File "lane_change_simulation.py" creates a car with a target speed of 90km/h behind main vehicle, does autopilot setup and agent creation. File "manual_control_lane_change.py" is the main file which contains logic to display a notification for a threat and it creates Pygame for human users to operate a vehicle and get a notification if there is any threat.

In Listing 5, there are important snippets related to lane change simulation. Every vehicle will parse BSM message from other vehicles and save their information in the dictionary. The queue data structure is used to save the 5 latest positions of other vehicles. Five latest locations and the latest speed are saved, and then based on that information car gets a notification if there is any threat from behind.

Listing 5: SPADE Agent Lane Change

```python
        def create_bsm_message(self):
            msg = spade.message.Message()
            if not self.agent.carla_vehicle:
                msg.body = ""
            if not self.agent.carla_vehicle.is_alive:
                raise VehicleDestroyed("Please stop agent")
            location = self.agent.carla_vehicle.get_transform().location
            crash = False
            if hasattr(self.agent.carla_vehicle, "crash") and self.agent.
             carla_vehicle.crash:
                crash = True
                self.agent.carla_vehicle.crash = False
            msg.set_metadata("performative", "inform")
            msg.set_metadata("ontology", "bsm")
            msg.body = f"{self.agent.name}*Location:{location.x},{location.y},{
             location.z}*Crash:{crash}*Speed:{get_speed(self.agent.carla_vehicle)}"
            return msg

        async def send_message_to_all(self, msg):
            for a in self.agent.receivers:
                msg.to = a
                await self.send(msg)

    class ParseBSM(spade.behaviour.CyclicBehaviour):
        async def run(self):
            msg = await self.receive(timeout=10)
            if msg:
                msg_splitted = msg.body.split("*")
                # print(f"msg_spliited: {msg_splitted}")
                car_name = msg_splitted[0]
                car_location = msg_splitted[1].split(":")[1]
```

```
                x,y,z = car_location.split(",")
                car_speed = float(msg_splitted[3].split(":")[1])
                loc = carla.Location(x=float(x), y=float(y), z=float(z))
                if car_name in self.agent.dict_positions:
                    self.agent.dict_positions[car_name].append(loc)
                else:
                    self.agent.dict_positions[car_name] = deque(maxlen=5)
                    self.agent.dict_positions[car_name].append(loc)
                self.agent.dict_vehicle_speed_list[car_name] = car_speed
                # print(f"{self.agent.name} got message: {msg.body}")

    async def setup(self):
        self.add_behaviour(self.SendBSMBehaviour(period=0.5), None)
        self.add_behaviour(self.ParseBSM(), bsm_template)
```

In Listing 6, there is a snippet of code that will try to determine if there is any threat from behind, and if that is the case, have a notification.

Listing 6: Threat Recognition Snippet

```
def getting_closer(positions, my_position):
    distance = 10000
    count = 0
    for pos in positions:
        dist_to_compare = pos.distance(my_position)
        if dist_to_compare < distance:
            distance = dist_to_compare
            count += 1
    if count >= 4:
        return True
    else:
        return False

def get_vehicle_approaching_status(my_speed, my_position):
    global dict_vehicle_speed_list
    for key in dict_vehicle_speed_list:
        if dict_vehicle_speed_list[key] > my_speed and getting_closer(dict_positions
         [key], my_position):
            return "WARNING - CAR IS GETTING CLOSE"
    return "No threat"
```

In Figure 14, there is a notification that there is a threat behind the target vehicle because a car is approaching at a greater speed than the current speed of the target vehicle.

In Figure 15, the threat is still active, and the car is much closer.

In Figure 16, there is no more threat, and the notification is not active anymore, a car is in front of the target vehicle.

The simulation was successful in representing how V2V communication can be used to notify the human operators that the threat is behind and that lane change might not be safe. While driving human operator can get a notification on dashboard that there is a thread behind to be careful when changing lanes. Notification can be implemented by manufacturers in

```
Server:              103 FPS
Client:               62 FPS

Vehicle:     Tesla Cybertruck
Map:                   Town04
Simulation time:      0:00:51

Speed:                46 km/h
Compass:              182° SW
Location:(-515.1,  228.3,  91.7)
GNSS:  (-0.002060, -0.004628)
Height:                  -0 m
Vehicle approaching: WARNING - CAR IS GETTING CLOSE

Throttle:
Steer:
Brake:
Reverse:     □
Hand brake:  □
Manual:      □
Gear:        1

Collision:

Number of vehicles:         2
Nearby vehicles:
  91m Seat Leon
```

Crossed line 'Broken'

Figure 14: Lane Change Notification Car is Far Away

```
Server:              101 FPS
Client:               62 FPS

Vehicle:     Tesla Cybertruck
Map:                   Town04
Simulation time:      0:00:57

Speed:                16 km/h
Compass:              169° SE
Location:(-513.8,  266.8,  78.5)
GNSS:  (-0.002406, -0.004614)
Height:                  -0 m
Vehicle approaching: WARNING - CAR IS GETTING CLOSE

Throttle:
Steer:
Brake:
Reverse:     □
Hand brake:  □
Manual:      □
Gear:        1

Collision:

Number of vehicles:         2
Nearby vehicles:
  4m Seat Leon
```

Crossed line 'Solid'

Figure 15: Lane Change Notification Car is Close

multiple ways: vibrating steering wheel, sound notification, visual notification, or combinations. If a car has self-drive capabilities it can even prevent a driver from steering to the left until the driver overrides.

```
Server:                91 FPS
Client:                62 FPS

Vehicle:     Tesla Cybertruck
Map:                   Town04
Simulation time:      0:00:58

Speed:                11 km/h
Compass:              166° SE
Location:(-512.1,  272.4,   75.7)
GNSS:  (-0.002456, -0.004598)
Height:                  -0 m
Vehicle approaching: No threat

Throttle:
Steer:
Brake:
Reverse:      ☐
Hand brake:   ☐
Manual:       ☐
Gear:         1

Collision:

Number of vehicles:         2
Nearby vehicles:
  31m Seat Leon
```

Crossed line 'Solid'

Figure 16: Lane Change Notification Car is In Front

5.5. Road Congestion Improvement Simulation by Using Infrastructure Messages

To simulate how V2X communication can improve road congestion good example is semaphore and vehicles in big cities. Every time there is a green light, there is a snake effect where the first car starts, then the second car starts, and so on. That creates a lot of lag which propagates. To improve this scenario traffic light is sending messages, and as soon as the target car gets a message for a green light, it stars automatically. If all other cars do the same there is a huge benefit. That example was simulated in the following way.

There is a new SPADE file named "semaphore_simulation_spade.py" that now contains the Semaphore agent. In Listing 7 there can be seen essential code snippets to achieve the described behavior. Semaphore will send its current state and location. Using that information car can decide if that semaphore is vital to him.

Listing 7: SPADE Semaphore agent

```
class SemaphoreAgent(spade.agent.Agent):
    def __init__(self, jid, password, semaphore_obj):
        super().__init__(jid, password)
        self.semaphore = semaphore_obj
        self.receivers = []

    class SendLightStateBehaviour(spade.behaviour.PeriodicBehaviour):
        async def run(self):
            try:
                msg = self.create_semaphore_message()
                if msg.body == "":
                    return
```

```
            except:
                await self.agent.stop()
                return
            await self.send_message_to_all(msg)

        async def on_end(self):
            pass

        async def on_start(self):
            pass

        def create_semaphore_message(self):
            msg = spade.message.Message()
            if not self.agent.semaphore:
                msg.body = ""
            location = self.agent.semaphore.get_transform().location
            msg.set_metadata("performative", "inform")
            msg.set_metadata("ontology", "environment")
            msg.body = f"Semaphore:{self.agent.semaphore.get_state()},{location.x},{location.y}"
            #print(msg.body)
            return msg

        async def send_message_to_all(self, msg):
            for a in self.agent.receivers:
                msg.to = a
                await self.send(msg)

    async def setup(self):
        self.add_behaviour(self.SendLightStateBehaviour(period=0.5), None)
```

Modification to CarAgent can be seen on Listing 8. It uses a really naive way to check if a semaphore is relevant to him. This approach can not be used in production, and it is only executed this way to simplify simulation.

Listing 8: CarAgent Modification for V2X

```
self.add_behaviour(self.ParseEnvMsg(), environment_template)
    class ParseEnvMsg(spade.behaviour.CyclicBehaviour):
        async def run(self):
            msg = await self.receive(timeout=10)
            if msg:
                if "Semaphore:Green" in msg.body:
                    msg_splitted = msg.body.split(",")
                    semaphore_location = carla.Location(x=float(msg_splitted[1]), y=float(msg_splitted[2]), z=0)
                    vehicle_location = self.agent.carla_vehicle.get_transform().location
                    print(f"{self.agent.name} - distance: {vehicle_location.distance(semaphore_location)}")
                    print(f"{self.agent.name} - stationary: {is_vehicle_stationary(self.agent.carla_vehicle)}")
                    if is_vehicle_stationary(self.agent.carla_vehicle) and
```

```
                 vehicle_location.distance(semaphore_location) < 90:
                    print("starting")
                    self.agent.carla_vehicle.set_autopilot(False)
                    control = carla.VehicleControl(throttle=1.0, steer=0.0,
                     brake=0.0)
                    self.agent.carla_vehicle.apply_control(control)
                    await asyncio.sleep(4)
                    self.agent.carla_vehicle.set_autopilot(True)
```

To be able to execute this simulation, two new files were created. File "semaphore_simulation.py" creates all the vehicles, does the camera setup and autopilot setup. File "semaphore_control.py" attaches SPADE agent to the target semaphore and gives control to a human operator to manually choose when the semaphore is red or green for easier control of the simulation. On Listing 9, there is a snippet that allows control through the terminal.

Listing 9: User Input Semaphore State

```
while True:
    command_from_user = await ainput("Enter ’g’ for green light, ’r’ for red
     light and ’x’ for exit: ")
    if command_from_user == "g":
        target_semaphore.set_state(carla.TrafficLightState.Green)
    elif command_from_user == "r":
        target_semaphore.set_state(carla.TrafficLightState.Red)
    else:
        break
await agent.stop()
```

The simulation was successful in showing the difference in vehicle lag. In Figure 17, there is an initial state without using V2X communication.

Figure 17: Initial State Without V2X Communication

In Figure 18, 3 seconds after green light, we can notice only a few vehicles moved other vehicles are stationary.

Figure 18: 3 Seconds After Green Light Without V2X Communication

In Figure 19, there is a state after 6 seconds of green light and only a few vehicles managed to pass through until now. Most of the vehicles are still in front semaphore.

In Figure 20, there is an initial state, it should be really similar to the initial state without V2X communication.

In Figure 21, there is a state 3 seconds after the green light using V2X communication, the difference can already be seen, and more vehicles already moved.

In Figure 22, there is a state 6 seconds after the green light using V2X communication now the difference is even more significant, and a lot more vehicles passed the semaphore than in the picture without using V2X. The reader is strongly encouraged to go through the video in the appendices.

Figure 19: 6 Seconds After Green Light Without V2X Communication

Figure 20: Initial State With V2X Communication

Figure 21: 3 Seconds After Green Light With V2X Communication

Figure 22: 6 Seconds After Green Light With V2X Communication

6. Conclusion

The exploration of vehicle-to-vehicle (V2V) communication in this paper provides substantial insights into its applicability, challenges, and future potential. Spanning a wide range of topics, from methods and techniques used in the study to the specific use-cases of V2V, the investigation offers a comprehensive understanding of the subject matter.

Most notably, the successful execution of all three simulations lends strong support to the practical application of V2V communication. The simulations on crash avoidance, lane change assistance on highways, and road congestion improvement using infrastructure messages demonstrated the tangible benefits that V2V can bring to road safety and traffic management.

One of the most significant challenges in the implementation of vehicle-to-vehicle (V2V) communication lies in ensuring system interoperability and security. The fragmented nature of technologies like DSRC and C-V2X complicates the standardization of V2V systems across different manufacturers and regions. Moreover, safeguarding the privacy and data integrity in V2V interactions is a growing concern. These technical hurdles are further compounded by societal factors such as public acceptance and privacy concerns. Altogether, these challenges represent a complex array of obstacles that must be overcome to realize the full potential of V2V communication. In this paper while executing simulations there was less focus on real world production problems and most of the focus was on tangable scenarios to even see how much can V2V or V2X communication improve road conditions.

V2V communication presents promising prospects for enhancing vehicular safety and traffic efficiency. While challenges exist, the successful simulations indicate a bright future and demand further investigation to accelerate the real-world application of these technologies.

Bibliography

[1] R. Daddanala, V. Mannava, ai Tawlbeh, and M. Al-Ramahi, “Vehicle to Vehicle (V2V) Communication Protocol: Components, Benefits, Challenges, Safety and Machine Learning Applications,” 2021.

[2] NHTSA. “Vehicle-to-Vehicle Communication.” (), [Online]. Available: `https://www.nhtsa.gov/technology-innovation/vehicle-vehicle-communication`.

[3] K. Aries. “What Is V2V Technology?: V2V vs V2I vs V2X Technology Systems.” (2021), [Online]. Available: `https://www.verizonconnect.com/resources/article/connected-vehicle-technology-v2v-v2i-v2x/` (visited on 07/09/2023).

[4] P. Jackson. “V2X explained: what is vehicle-to-everything communication and how does it work? – Drive Section.” (2020), [Online]. Available: `https://www.drivesection.com/tech/vehicle-to-everything-explained/` (visited on 08/06/2023).

[5] M. Lotman. “Car Talk: Your Vehicle is About to Get Chatty | The Technoskeptic.” (2015), [Online]. Available: `https://thetechnoskeptic.com/car-talk/` (visited on 08/06/2023).

[6] J. Jermann. “Truck Platooning | Rapp AG.” (), [Online]. Available: `https://www.rapp.ch/de/stories/truck-platooning` (visited on 08/06/2023).

[7] H. A. Ameen, A. K. Mahamad, S. Saon, A. Kadir Mahamad, D. M. Nor, and K. Ghazi, “A review on vehicle to vehicle communication system applications hearing model and its application View project room material View project A review on vehicle to vehicle communication system applications,” *Article in Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 1, pp. 188–198, 2020, ISSN: 2502-4752. DOI: `10.11591/ijeecs.v18.i1.pp188-198`.

[8] F. Markus. “Feds Finally Get Out of the Way to Enable Cellular V2X Communication.” (2023), [Online]. Available: `https://www.motortrend.com/news/fcc-ruling-dsrc-5g-v2x-communication-technology/` (visited on 07/09/2023).

[9] M. Lowenstein. “What’s the state of C-V2X? — Lowenstein.” (2023), [Online]. Available: `https://www.fiercewireless.com/wireless/whats-state-c-v2x-lowenstein` (visited on 07/09/2023).

[10] “Vehicle to Vehicle Communication Market Size, Trends and Global Forecast To 2032.” (), [Online]. Available: `https://www.thebusinessresearchcompany.com/report/v2v-communication-global-market-report` (visited on 07/09/2023).

[11] J. Koon. “Competing V2V Technologies Emerge, Create Confusion.” (2023), [Online]. Available: `https://semiengineering.com/competing-v2v-technologies-emerge-create-confusion/` (visited on 07/09/2023).

[12] R. K. Jurgen, *V2V/V2I Communications for Improved Road Safety and Efficiency*. 2012, ISBN: 9780768078053. DOI: `10.4271/pt-154`.

[13] J. Harding, G. Powell, R. Yoon, *et al.*, “Vehicle-to-vehicle communications : readiness of V2V technology for application.,” 08/2014. DOI: `10.21949/1503647`.

[14] Car Bike Tech. “What is Vehicle to Vehicle OR V2V Communication Technology?” (2015), [Online]. Available: `https://carbiketech.com/vehicle-to-vehicle-v2v-communication/` (visited on 07/09/2023).

[15] M. Uzair, “Vehicular Wireless Communication Standards: Challenges and Comparison 379 Review Paper,” 2014.

[16] G. Dimitrakopoulos, *Current Technologies in Vehicular Communication*. Springer, 2016, ISBN: 9783319472430.

[17] H. P. D. Nguyen and R. Zoltán, “The Current Security Challenges of Vehicle Communication in the Future Transportation System,” *SISY 2018 - IEEE 16th International Symposium on Intelligent Systems and Informatics, Proceedings*, pp. 161–165, 11/2018. DOI: `10.1109/SISY.2018.8524773`.

[18] AutoCrypt. “AutoCrypt PKI | Secure Authentication for Connected Mobility.” (), [Online]. Available: `https://autocrypt.io/products/pki/` (visited on 07/11/2023).

[19] SAE International, “SURFACE VEHICLE STANDARD - SAE Standard,” *SAE International*, vol. 4970, no. 724, pp. 1–5, 2022.

[20] F. Ahmed-zaid and F. M. Company, “SAE Standard Overview of Basic Safety Message (BSM) - Ford,” 2019.

[21] F. Hu, *VEHICLE-TO-VEHICLE AND VEHICLE-TO-INFRASTRUCTURE COMMUNICATIONS : a technical.* CRC PRESS, 2018, ISBN: 9780367572020.

[22] “CARLA Simulator.” (), [Online]. Available: `https://carla.readthedocs.io/en/latest/` (visited on 07/14/2023).

[23] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *Proceedings of the 1st Annual Conference on Robot Learning*, California, USA, 2017, pp. 1–16.

[24] Spade. “SPADE’s documentation! — SPADE 3.3.0 documentation.” (), [Online]. Available: `https://spade-mas.readthedocs.io/en/latest/` (visited on 07/14/2023).

[25] “ejabberd XMPP Server with MQTT Broker and SIP Service.” (), [Online]. Available: `https://www.ejabberd.im/` (visited on 08/22/2023).

[26] M. Schatten and B. Okreša Ðurić, *Priručnik za laboratorijske vježbe iz Višeagentnih sustava*, Varaždin: 2022.

[27] M. Wooldridge, *An Introduction to MultiAgent Systems*, Second edi. John Wiley and Sons, 2009.

List of Figures

List of Listings

Appendices

1. Link to the GitHub Repository of the Project

Link to the GitHub repository of the project: https://github.com/markoBel3/v2v-simulation

2. Link to the video of Crash Prevention Simulation

Link to the video of Crash Prevention Simulation: https://youtu.be/GgEwg76rXqQ

3. Link to the video of Lane Change On Highway Assist

Link to the video of Lane Change On Highway Assist Simulation: https://youtu.be/WIjcTyJVGSE

4. Link to the video of Road Congestion Improvement Simulation by Using Infrastructure Messages

Link to the video of Road Congestion Improvement Simulation by Using Infrastructure Messages: https://youtu.be/TMGYD3eYedQ