D5.3

Protocols for reliable V2X message exchange

Project Acronym	TransAID	
Project Title	Transition Areas for Infrastructure-Assisted Driving	
Project Number	Horizon 2020 ART-05-2016 – GA Nr 723390	
Work Package	WP5	Connectivity and Signalling
Lead Beneficiary	Universidad Miguel Hernández (UMH)	
Editor / Main Author	Baldomero Coll-Perales Gokulnath Thandavarayan	UMH
Reviewer	Anton Wijbenga	MAPtm
Dissemination Level	PU	
Contractual Delivery Date	29/02/2020 (M30)	
Actual Delivery Date	18/03/2020	
Version	v1.0	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723390.

Document revision history

Version	Date	Comments
v0.1	10/01/2020	Initial draft version
v0.2	14/01/2020	First draft of study on broadcast acknowledgment included
v0.3	20/01/2020	First draft of study on impact of DCC on V2X applications reliability included
v0.4	03/02/2020	First draft of study on compression of V2X messages included
v0.5	14/02/2020	Extension of the study on the impact of DCC on V2X applications by including the results of the Facilities DCC mechanism with feedback
v0.6	20/02/2020	Extension of the study on compression of V2X messages by adding the results obtained through the compression of real CAM messages obtained using TransAID's prototypes.
v0.6	25/02/2020	Extension of the study on compression of V2X messages by adding the results obtained through the compression of real CPM and MCM messages obtained using TransAID's prototypes.
v0.7	06/03/2020	Complete version for internal review
v1.0	18/03/2020	Final version

Editor / Main author

Baldomero Coll-Perales (UMH) / Gokulnath Thandavarayan (UMH)

List of contributors

Baldomero Coll-Perales (UMH), Gokulnath Thandavarayan (UMH), Miguel Sepulcre (UMH), Javier Gozalvez (UMH), Alejandro Correa (UMH)

List of reviewers

Anton Wijbenga (MAPtm), Julian Schindler (DLR)

Dissemination level:

■ PU : Public

□ RE : Restricted to a group specified by the consortium (including the Commission Services)

□ CO : Confidential, only for members of the consortium (including the Commission Services)

Table of contents

Document revision history ... 2
Table of contents ... 3
Executive Summary ... 5
1 Introduction ... 6
1.1 TransAID focus and objective ... 6
1.2 The TransAID iterative approach ... 7
1.3 Purpose of this document ... 7
1.4 Structure of this document ... 9
1.5 Glossary ... 9
2 Interference Mitigation through V2X Message Compression ... 12
2.1 Data compression ... 13
2.1.1 Entropy compression ... 13
2.1.2 Adaptive dictionary compression ... 15
2.2 Analysis of V2X messages ... 16
2.2.1 Messages structure and format ... 16
2.2.1.1 CAM ... 16
2.2.1.2 CPM ... 16
2.2.1.3 MCM ... 17
2.2.2 Compression bounds ... 18
2.3 Compression of V2X messages ... 22
2.4 Communications performance ... 25
3 Congestion control for enhanced V2X reliability ... 29
3.1 ETSI DCC Framework ... 29
3.2 Access Layer DCC ... 31
3.2.1 Evaluation ... 32
3.2.1.1 Scenario and settings ... 32
3.2.1.2 Results ... 34
3.3 Facilities Layer DCC ... 55
3.3.1 ETSI solution ... 55
3.3.2 TransAID proposal ... 57
3.3.3 Evaluation ... 58
4 Context-based Acknowledgement for Cooperative Broadcast V2X messages ... 65
4.1 Context-based Broadcast Acknowledgment ... 67

4.1.1 Concept 67
4.1.2 Proposal 68
4.2 Scenario of Evaluation 70
4.2.1 Simulation environment 70
4.2.2 CPM's content and generation rules 72
4.3 Results 73
5 Conclusion 80
References 81

Executive Summary

The TransAID (Transition Areas for Infrastructure-Assisted Driving) project is defining traffic management manoeuvres to help connected and automated vehicles (CAVs) to deal with traffic conditions or zones that their automated systems are not able to handle by themselves. To this aim, TransAID leverages vehicles' V2X connectivity technology to implement traffic management measures that are based on the exchange of information between vehicles (V2V) and between vehicles and the infrastructure (V2I). V2X communications, in general, use broadcast transmissions to inform surrounding vehicles. This might include, for example, the vehicles' status information or the occurrence of dangerous situations through the Cooperative Awareness (CA) and Decentralised Environment Notification (DEN) services, respectively. Examples of more recent services are the novel Collective Perception (CP) and Manoeuvre Coordination (MC) services through which vehicles and the infrastructure share information about detected objects and planned trajectories among others, respectively.

All these V2X applications rely on broadcast messages and lack mechanisms to ensure the correct delivery of messages. In this context, TransAID aims to improve the reliability of V2X communications. This Deliverable includes the TransAID contributions that have addressed this issue at different and complementary layers of the V2X communication stack.

First, TransAID has studied and evaluated the benefits that compressing the V2X messages bring to the reliability of the V2X communications. The study has shown the feasibility of compressing real-world CAM, CPM and MCM messages, and that the considerable reduction of the messages' size result in a reduction of the channel load and interferences. This is key to improve the reliability of V2X communications without modifying the amount of information sent.

Second, TransAID has analysed the impact of the ETSI's Decentralised Congestion Control (DCC) mechanism on the V2X messages reliability. ETSI's DCC is designed to reduce the rate of messages that V2X applications can transmit with the aim to keep the channel load below a threshold. To this aim, DCC enqueues messages and transmits them when it is allowed. Messages on queue can be dropped or a queue overflow can happen, which leads to negative effects to the V2X applications reliability. TransAID has demonstrated the reliability benefits of linking the generation of messages by the V2X applications to the DCC transmission rate.

Finally, TransAID has proposed a novel mechanism that adds acknowledgements to broadcast V2X messages. The solution proposed by TransAID is designed to take into account the scalability issues that acknowledgments can provoke to broadcast transmissions. In particular, the TransAID proposal establishes that: 1) only some critical V2X messages have to be acknowledged; and 2) the acknowledgement is only required to few receivers. These two aspects are identified at the V2X application layer and notified to the bottom layers so that the required message exchange is triggered.

Without loss of generality, since this can be applied to any V2X broadcast message, TransAID has demonstrated the benefits of the proposed solution to improve the reliability of CPM messages at an intersection scenario.

1 Introduction

This section presents a concise overview of the TransAID project followed by the purpose and structure of the document.

1.1 TransAID focus and objective

Automated Vehicles (AVs) have potential to improve traffic efficiency and road safety by applying automatization of perception and control tasks aimed at going beyond the capabilities of human drivers. A growing number of useful automation features in production cars is a visible trend already today. Moreover, first examples of highly and full automated driving have been showcased to work on real roads under specific conditions (e.g. highway scenarios) and are about to go on the market [1][2][3][4][5]. In this context, the automotive industry is spending efforts towards preparing future highly and fully AVs to support an increasing number of road conditions and traffic situations. Despite these efforts, different studies have shown that automated driving will not always be possible: whenever for the different reasons listed in [6] automated systems reach their functional limits and are not able to handle a traffic situation by their own, they will require a Transition of Control (ToC) to manual driving (downward ToC in this case). As the system detects that automation can be resumed, an upward ToC to automated driving is triggered. In case of a downward ToC, the driver needs some time to become familiar with the situation to take over safely. This time (i.e. the duration of a safe ToC) generally increases with higher levels of automation, especially if the driver is distracted [7] and as the traffic density increases [8].

If the driver is not responding to a ToC request, an automated vehicle shall try to perform a so called Minimum Risk Manoeuvre (MRM) to bring the vehicle into a safe state (e.g. decelerating to a full stop, or change lanes to occupy a safe spot [9]). As it can be imagined, besides affecting the safety of an AV driver, a ToC might also negatively impact the safety of surrounding traffic participants and compromise the traffic flow, especially in so called “Transition Areas” where multiple ToCs can occur simultaneously.

In this context, the TransAID project aims at developing and testing traffic management measures to eliminate or mitigate the negative effects of ToCs in Transition Areas. The TransAID traffic management measures shall be designed to operate in future mixed traffic scenarios where automated, cooperative, and conventional vehicles will coexist. To this aim, TransAID extends conventional signalling means by cooperative ITS (C-ITS) and V2X communications, whose deployment at both vehicles and road infrastructure in Europe is about to start as a “Day1” phase in 2019, as stated by the industrial organization Car2Car Communication Consortium (C2C-CC) and the road infrastructure platform C-Roads [10].

In TransAID, infrastructure and vehicles will use C-ITS to improve their perception and knowledge of the environment. In fact, V2X communications from different categories of cooperative vehicles (automated and non-automated) will allow the road infrastructure to perform a more precise and real-time assessment of traffic demands and stream composition (i.e. share of different vehicle categories according to the TransAID vehicle classification of [11]). Once this information is obtained, V2X communications will be used by the road infrastructure to inform about warnings and suggest manoeuvres that, when implemented by the addressed vehicles, will help managing traffic situations associated to possible ToCs more effectively. These suggestions will be either implemented directly by vehicles (e.g. performing a ToC at a given position and time) or will be the triggering condition for the execution of cooperative manoeuvre by Cooperative Automated vehicles (CAVs). In fact, in of conflicting situations (e.g. a CAV is suggested to move onto a lane where another CAV is following with higher speed), CAVs will start using V2X communications to negotiate the right of way, hence ensuring safety without hampering the traffic flow.

1.2 The TransAID iterative approach

As mentioned above and better detailed in deliverable D2.1 [6], TransAID measures are designed to follow a hierarchical approach where control actions are implemented at different layers including centralised traffic management, infrastructure, and vehicles. TransAID therefore takes into account a foreseen mix of conventional/legacy vehicles (LV), cooperative non-automated vehicles (CV), non-cooperative automated vehicles (AV) and cooperative automated vehicles (CAV). The infrastructure will integrate the acquired information at the Traffic Management System (TMS). The TMS will generate progression plans which are taken over by the infrastructure and communicated to vehicles, either by V2X communication from the Roadside Infrastructure (RSI) or by e.g. variable message signs (VMS) for reaching non-equipped vehicles (LV/AV).

To validate the effectiveness of its management measures, TransAID adopts simulations taking into account traffic safety and efficiency metrics. For the simulations to be as reliable as possible, the most relevant microscopic traffic models for mixed traffic behaviour and interactions with AD cars are developed. Also, communication protocols for the cooperation between CAVs, CVs, and the cooperative RSI are implemented, modelled and included. Based on the results of these simulations, the most promising solutions are then implemented as real world prototypes and demonstrated in closed and controlled environments as proof of concepts for technical feasibility.

The above-mentioned approach is applied over two iterations, each taking half of the project’s total duration. During the first iteration, the focus is on studying aspects of transition of control and transition areas through basic scenarios. This implies that realistic models for AD and communication protocols need to be developed and/or adopted to cover the requirements of these scenarios’ simulations. Using the basic scenarios, it is possible to run many simulations and focus in detail on the relatively new aspects of ToC, Transition Areas (TAs) and measures mitigating negative effects of ToC. The goal of the first iteration is hence to gain experience with all aspects relevant to TAs and mitigating measures. In the second iteration, the achieved experience is used to improve/extend the traffic management measures while at the same time increasing the complexity of the investigated scenarios (e.g. including more challenging scenarios not considered in the first iteration, or combining multiple scenarios in the same evaluation). The second iteration will consequently need additional functionalities from the traffic and communication protocols point of view, whose modelling will be implemented at later stages.

1.3 Purpose of this document

TransAID’s WP5 has defined the V2X message sets that specify the information exchanged between connected/automated vehicles and the infrastructure to improve the traffic-flow and efficiency, and support the situations where automated vehicles might need to perform a Transition of Control. Leveraging this V2X message set, TransAID has designed novel cooperative communication protocols that enable advanced sensing and perception mechanisms. It also has defined the V2X message flow between connected/automated vehicles and the infrastructure to implement cooperative manoeuvres, especially in identified Transition Areas. A key aspect for the proper operation of the designed cooperative V2X mechanisms is to ensure the reliable exchange of the V2X messages between the vehicles, and between the vehicles and the infrastructure. In this context, TransAID has identified and designed innovative mechanisms to successfully and efficiently improve the reliability of the V2X message exchange. An important aspect in this regard is that TransAID has addressed this issue at different and complementary levels, including the C-ITS Application, Transport, Networking and Access layers.

First, TransAID has addressed one of the most common issues affecting the V2X communications reliability that is the interfere caused between the communication parties. This is commonly addressed by modifying the transmission parameters in order to create more robust signals that can be received even under the presence of high interference. TransAID proposes a novel approach of addressing the interference issue without compromising the transmission rate which is usually the price to pay when more robust signals are transmitted. In particular, TransAID proposes to compress the V2X messages to be transmitted in order to reduce their size and therefore the channel load they generate. This compression is to be applied in a middleware layer between the C-ITS Facilities and Transport layers, i.e. compliant with the current standard processes. Obviously, the benefits of compressing the V2X messages depend on factors such as the compression gain and the message size. In this context, TransAID has studied how much the V2X messages can be compressed, and how it affects to the communications reliability. TransAID studies have shown that this depends on the compression techniques utilised (e.g. fixed dictionary vs adaptive) and the properties of the V2X messages (e.g. entropy). These studies have been conducted using TransAID standard V2X CAM, CPM and MCM messages obtained during the field tests of WP7 [12].

Second, TransAID has analysed the impact of the ETSI's decentralised congestion control (DCC) mechanism at the Access layer on the V2X messages reliability. ETSI's DCC Access is actually designed to reduce the channel congestion which might therefore contribute to improve the V2X message reliability. To this aim, the ETSI's DCC Access seeks distributing the channel usage among the neighbouring vehicles within the same radio communication area. The channel usage distribution is achieved, for example, by limiting the rate of V2X messages that each vehicle can transmit. DCC Access implements queues were V2X messages wait until they are allowed to be transmitted. However, this approach might have negative effects if it is not properly linked to the generation of V2X messages at the application layer, since V2X packets on the queues could be dropped because of a queue overflow or because their *time to live* could expire while they are waiting to be transmitted. TransAID has then analysed the impact of the ETSI's DCC on the reliability V2X messages, and in particular on the performance of the collective perception service that relies on the successful and timely delivery of CPM messages. To improve the reliability of the V2X message exchange, TransAID has also evaluated and evolved the Facilities DCC being standardized by ETSI.

Finally, TransAID has addressed the reliability of V2X communications by designing a mechanism to ensure the correct delivery of the messages. Most of the V2X applications/services are supported by the continuous exchange of V2X broadcast messages, and there is not a mechanism to inform the transmitter whether the transmission was correctly received by the receiver(s) or not. This is the case because broadcast V2X transmissions are not addressed to a particular vehicle, but to the set the of vehicles within the V2X communication range. The use of acknowledgments, as it is the case in unicast transmissions, might have scalability issues which could result in an uncontrolled flood of messages from the receiver(s) to the transmitters. In this context, TransAID has proposed a mechanism to address scalability while addressing reliability of broadcast V2X transmissions. The proposed mechanism is based on a cross-layer approach that first acts at the application layer by identifying the V2X messages that need to be acknowledged. At the application layer, the transmitting vehicle also identifies the receiving vehicle(s) that has (have) to report with an acknowledgment whether the transmission was correctly received or not. This triggers the generation of a separate unicast packet to request the identified receiving vehicle(s) to transmit such acknowledgment. The proposed mechanism also implements retransmissions of such specific/critical V2X messages if they are not correctly received. TransAID has demonstrated the effectiveness of the proposed mechanism e.g. at an intersection scenario where CPM messages are used to support the awareness of vehicle automations when pedestrians cross the street.

1.4 Structure of this document

The rest of this document is organized as follows. Section 2 describes the study conducted in TransAID to analyse the benefits of the V2X messages compression on the communications reliability. Section 3 focuses on the study of the impact of ETSI's DCC congestion control mechanism on the reliability and timeliness of CPM messages. Section 4 shows and analyses the mechanism proposed in TransAID that is based on the use of acknowledgments of selective/critical broadcast V2X messages. Section 5 indicates what are the main conclusions reached in this deliverable.

1.5 Glossary

Abbreviation/Term	**Definition**
ACK	Acknowledgement
A-MPDU	Aggregated MAC Packet Data Unit
A-MSDU	Aggregated MAC Service Data Unit
BAR	Broadcast ACK Request
C-ITS	Cooperative Intelligent Transport Systems
C2C-CC	Car2Car Communication Consortium
CA	Cooperative Awareness
CAM	Cooperative Awareness Message
CAV	Cooperative Automated Vehicle
CBR	Channel Busy Ratio
CD	Critical Distance
CR	Critical Range
CP	Collective Perception
CPM	Collective Perception Message
CPS	Collective Perception Service
DCC	Decentralized Congestion Control
DCC_ACC	DCC Access
DCC_CROSS	DCC Cross layer

DCC_FAC	DCC Facilities
DCC_NET	DCC Network
DE	Data Element
DEN	Decentralized Environmental Notification
DENM	Decentralized Environmental Notification Message
DF	Data Frame
DX.X	Deliverable X.X
EDCA	Enhanced Distributed Channel Access
ETSI	European Telecommunication Standards Institute
FoV	Field of View
FIFO	First In First Out
HARQ	Hybrid Automatic Repeat Request
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
ITS	Intelligent Transport System
ITS-G5	Access technology to be used in frequency bands dedicated for European ITS
LDPC	Low Density Parity Check
LIDAR	Light Detection and Ranging/Laser Imaging Detection and Ranging
LoS	Line of Sight
LTE-V	Long Term Evolution - Vehicular
MAC	Medium Access Control layer
MCM	Maneuver Coordination Message
MCS	Manoeuvre Coordination Service
NACK	Non-ACK
NGV	Next Generation Vehicular networks
NLoS	Non-LoS

NR	New Radio
OSI	Open Systems Interconnection
PDF	Probability Density Function
PDR	Packet Delivery Ratio
PDU	Protocol Data Unit
POC	Perceived Object Containers
PHY	Physical layer
QPSK	Quadrature Phase Shift Keying
RSU	Road Side Unit
RT	Reaction Time
SIFS	Short Inter Frame Space
ToC	Transition of Control
TransAID	Transition Areas for Infrastructure-Assisted Driving
TTL	Time to Live
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-anything
VRU	Vulnerable Road Users
WP	Work Package

2 Interference Mitigation through V2X Message Compression

The reliability of the V2X message exchange significantly depends on the channel load and interferences in the network. To reduce the load and interferences, different congestion control protocols have been proposed to date. One of the most relevant ones is the DCC (Decentralized Congestion Control) solution defined by ETSI in Europe that spans over multiple layers of the protocol stack [13] (see Section 3 of this document for more details). The solutions proposed to date normally adapt the communication parameters, e.g. transmission power, the message transmission frequency or the data rate to control the load. This type of solutions can influence the application's effectiveness, since they affect e.g. the communication range.

Data compression is an alternative approach to reduce the channel load and improve the reliability of vehicular networks that would not require the modification of the communication parameters. Data compression is widely used in communication systems to improve the bandwidth utilization. For example, in HTTP, data is compressed before it is sent from servers. Browsers are in charge of downloading and decompressing the received data. The most common compression schemes used for HTTP compression include Gzip and Compress [14]. According to [15], HTTP compression can provide a compression gain of around 75% for text files (HTML, CSS, and JavaScript).

In TransAID, we have proposed and explored for the first time the use of data compression to reduce the channel load and improve the reliability of the V2X message exchange in vehicular networks. We propose that the payload of V2X messages is compressed after being generated by the upper layers to reduce the number of bits of each message without reducing the amount of information to be sent. This could be implemented at the Facilities layer of the ETSI (see Figure 1) or ISO ITS Architectures, or above the WSMP Transport Layer of the 1609/WAVE Architecture. At the receiver, compressed V2X messages will need to be decompressed to recover the original messages. A new module could be incorporated to the protocol stack in all vehicles for the data compression and decompression. The proposed data compression would not require any other significant modification of the protocol stack, which increases its potential for standardization and real-world implementation. It would only require additional processing power and power consumption to enable the real time compression and decompression of the messages.

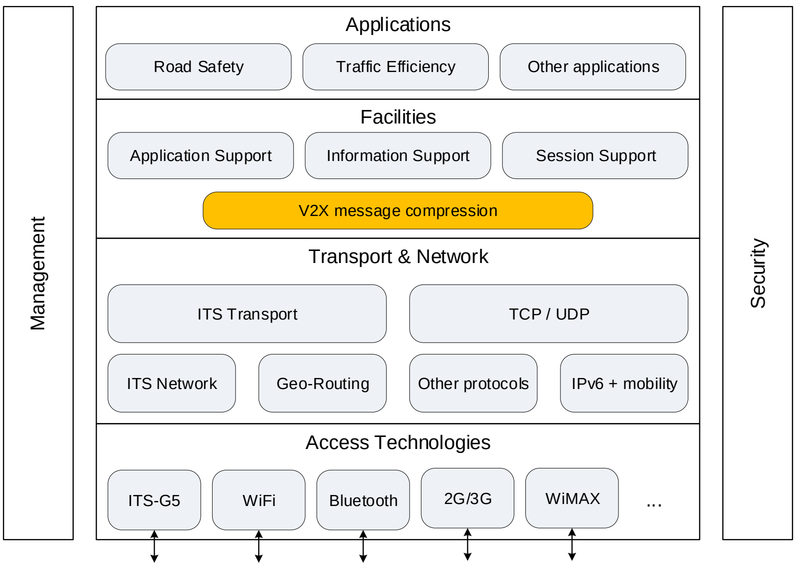


Figure 1: ETSI ITS architecture with proposed V2X message compression.

The compression gain depends significantly on the size of the input data and the type of data itself. For example, large text files normally have repeated substrings (e.g. words) and can be significantly compressed. This is the case because compression algorithms are often designed to replace repeated substrings with a pointer to the previous occurrence of the repeated substrings. However, V2X messages are characterized by having a relatively small size (hundreds of Bytes). In addition, the content of V2X messages can significantly vary and the existence of repeated substrings has not been studied yet.

In this deliverable, we provide first results of data compression and decompression in vehicular networks by evaluating the compression gain of different algorithms, and also the positive effects of the V2X message compression on the channel load and communications performance. In this study we apply the data compression to real-world CAMs (Cooperative Awareness Messages), CPMs (Collective Perception Messages) and MCMs (Manoeuvre Coordination Messages) generated in TransAID WP7 [12].

2.1 Data compression

Data compression is the process of encoding information to reduce the number of bits of the original representation and is also known as source coding in data transmission. Data compression can be either lossy or lossless. Lossy compression reduces the number of bits by removing unnecessary or less important information and is typically applied in audio or image compression. Lossless compression reduces bits by identifying and eliminating statistical redundancy. Lossless compression allows the original data to be perfectly reconstructed from the compressed data, i.e. without any loss due to the compression/decompression processes. The focus of this work will be therefore on lossless compression.

2.1.1 Entropy compression

The idea behind entropy compression (also known as probability compression) is to divide the message source in symbols of equal length. The set of possible symbols will be referred to as the alphabet. Frequent symbols are encoded in fewer bits than infrequent symbols. Entropy compression algorithms thus take into account the probability that each symbol appears in a message. Given the probabilities, a table of codewords or dictionary can be constructed. Codewords for symbols with low probabilities have more bits, and codewords for symbols with high probabilities have fewer bits.

To apply this type of algorithms, the probabilities of the symbols in the alphabet are needed to construct the dictionary. Different methods exist:

- Construct a different dictionary per V2X message individually. This solution requires calculating the probabilities of the different symbols for each V2X message before it is transmitted. This solution implies a different dictionary of codewords for each message. It ensures that the dictionary constructed for each message is the optimal one. However, the dictionary is needed in the decompression process and thus it requires the transmitter to send it to the receiver together with the message.
- Construct a dictionary per V2X message type based on a set of messages of each type. This solution is based on the analysis of a set of V2X messages of certain type (e.g. CAM, CPM or MCM) to extract the probabilities of the different symbols and construct one dictionary per V2X message type. If the dictionary is fixed and known by the transmitter and receiver, it does not need to be exchanged. The main drawback is that the dictionary is the optimum for the set of messages analysed but might not be optimal for each individual message.

In TransAID, we have adopted the second solution that constructs a dictionary per V2X message type, given that it does not require the exchange of the dictionary between transmitter and receiver.

The limits of data compression for entropy compression algorithms are established by the Shannon's source coding theorem. This theorem shows that it is impossible to compress the data such that the average number of bits per symbol is less than the Shannon entropy of the source. In other words, n independent and identically-distributed random variables each with entropy $H(X)$ can be compressed into more than $n \cdot H(X)$ bits with negligible risk of information loss [17]. This limit can only be reached in practice for large values of n (i.e. large V2X messages in our context). However, it can be useful in this study to estimate how close to the limit the algorithms and techniques here considered are.

To calculate the data compression limit, the Shannon entropy of a source, i.e. $H(X)$, needs to be defined. The entropy of a memoryless source can be defined as the average amount of information acquired by observation of a single symbol on the source output [18]. The amount of information acquired by observation of a given symbol a_i is given by the formula:

$$I(a_i) = \log_2\left(\frac{1}{P(a_i)}\right) \qquad (1)$$

where $P(a_i)$ is the probability that the source generates the symbol a_i. We will consider that the source has an alphabet of K possible symbols, $X = \{a_1, \ldots, a_K\}$, and each of them is generated with probability $P(a_1), \ldots P(a_K)$. Therefore, the Shannon entropy of a given source can be calculated as:

$$H(X) = E[I(a_i)] = \sum_{i=1}^{K} P(a_i)\log_2\left(\frac{1}{P(a_i)}\right) \qquad (2)$$

Without compression, each symbol could be represented by $log_2(K)$ bits, and therefore the number of bits needed to represent a V2X message with n symbols is:

$$B_{nc} = n \cdot \log_2(K) \qquad (3)$$

For example, if the alphabet contains K=16 symbols, each symbol can be represented by $log_2(16)$=4 bits, from 0000 to 1111; therefore, a V2X message with n=200 symbols can be represented by B_{nc}=800 bits without compression. Given Shannon's source coding theorem, the minimum number of bits of the message is bounded by:

$$B_c = n \cdot H(X) \qquad (4)$$

The compression gain limit of a V2X message with n symbols can be therefore expressed as a percentage using the following equation:

$$CG_{\lim}[\%] = 100 \cdot \frac{B_{nc} - B_c}{B_{nc}} = 100\left(1 - \frac{H(X)}{\log_2(K)}\right) \qquad (5)$$

As it can be observed, this limit does not depend on the length of the V2X message but depends on the number of symbols in the alphabet (K) and the probabilities of the different symbols ($H(X)$).

Different entropy compression algorithms have been proposed in the literature. One of the first well-known entropy compression algorithms is known as Shannon-Fano coding [18]. Assuming that the probability of each symbol in the alphabet is known, the Shannon-Fano coding algorithm is composed by the following steps:

1. Sort the list of symbols in decreasing order of probability, the most probable ones to the left and least probable to the right.
2. Split the list into two parts, with the total probability of both parts being as close to each other as possible.
3. Assign bit 0 to the left part and bit 1 to the right part.
4. Repeat the steps 2 and 3 for each part, until all the symbols are split into individual subgroups.

This algorithm is needed to build the dictionary of codewords. It only needs to be executed when the probabilities of the symbols of the alphabet change. We therefore assume that these probabilities are fixed for each message type and will build one dictionary for CAM, one for CPM and one for MCM. Once the dictionary of codewords is obtained, it is used to compress the messages by replacing each symbol with its codeword.

2.1.2 Adaptive dictionary compression

Adaptive dictionary compression considers a completely different approach to entropy compression. When using an adaptive dictionary compression algorithm, data do not have to be parsed before compressing in order to calculate the symbols probabilities. They achieve compression by looking for repeated substrings in the input data. To this aim, adaptive schemes start either with no dictionary or with a default baseline dictionary. As compression proceeds, the algorithm adds new symbols to the dictionary following certain rules. The input data is read to look for groups of symbols that appear in the dictionary. If a string match is found, a pointer or index into the dictionary is sent to the output instead of the code for the symbol. The longer the match, the better the compression ratio. Adaptive dictionary compression methods have become the de facto standard for general-purpose data compression due to their high-performance compression combined with reasonable memory requirements.

The two adaptive dictionary compression tools considered in this study are Compress [19] and Gzip [20]. Both tools are open source solutions and widely used. They both are based on the well-known Lempel-Zip algorithm.

Gzip makes use of the original Lempel-Ziv algorithm, also known as LZ77 [21]. The LZ77 algorithm looks for repeated substrings based on the concept of sliding window. As the data is compressed, LZ77 only looks for repeated substrings in a window of previously compressed data. The window is divided into a search buffer containing the data that has already been compressed, and a lookahead buffer containing the data yet to be compressed. As the data is compressed, the window slides along, removing the oldest compressed data from the search buffer and adding new uncompressed data to the lookahead buffer. Once a substring in the lookahead buffer is found to be completely contained in the search buffer, it is replaced by its position in the search buffer and its length. The output format produced by Gzip is described in RFC 1952 [22] and includes a 10-byte header, some optional extra headers, the compressed data and an 8-byte footer containing a CRC-32 checksum and the length of the original uncompressed data.

Compress uses the Lempel-Ziv-Welch algorithm or LZW [23]. The LZW algorithm makes use of a dictionary that is built based on the input data, where each entry in the dictionary has an index. If the algorithm is configured to operate using bytes, it is initialized with one entry for each of the 256 possible values. When a substring, *S*, of the data being analysed is found in the dictionary, it is replaced by its index and a new entry is added to the dictionary that contains *S* and the next symbol in the data. This means that new entries are only added if a prefix one byte shorter is already in the dictionary (e.g. “sun” is only added if “su” had previously appeared in the data).

The compression gain depends on the size of the input data and the distribution of common substrings. Typically, LZ77 (Gzip) is able to achieve a compression gain of 60-70% for text such as source code or English texts, while LZW (Compress) is able to achieve 50-60% also for text [19][20]. The compression gain that these algorithms could provide when compressing CAMs, CPMs and MCM, or other V2X messages, needs yet to be studied, because this significantly depends on their length and their content (e.g. existence of repeated sequences).

2.2 Analysis of V2X messages

The statistical properties and the size of the data to be compressed can have a high impact on the performance of the compression process. For this reason, this work considers standard-compliant V2X messages defined by ETSI, populated with real data in real-world experiments conducted in TransAID WP7 [12]. The focus is on CAM, CPM and MCM messages since they will be continuously transmitted by all vehicles and are expected to consume a high proportion of the channel bandwidth. Section 2.2.1 summarizes the structure and format of the different messages (more information can be found in [24]). Section 2.2.2 conducts a statistical analysis of the different message types considered, and derives the theoretical compression bounds for the entropy compression algorithms used.

2.2.1 Messages structure and format

2.2.1.1 CAM

A CAM is composed of one common header and multiple containers [25] (see Figure 2). The common header is known as ITS PDU header and includes the protocol version, the message type and the ID of the vehicle or RSU (Road Side Unit) that transmits the CAM. For vehicles, a CAM must contain one Basic Container and one High Frequency Container (both are mandatory) and may also include one Low Frequency Container and one or more other Special Containers. The Basic Container includes information related to the transmitting vehicle, such as the type of vehicle or its geographic position. The High Frequency Container contains highly dynamic information of the transmitting vehicle, such as its heading or speed. The Low Frequency Container contains static and not highly dynamic information of the transmitting vehicle, such as the status of the exterior lights. The Special Vehicle container includes information specific to the vehicle role. Each container is composed of a sequence of optional or mandatory data elements (DE) and/or data frames (DF). The length of a CAM depends on the number of optional containers considered. Since many containers, DEs and DFs are optional in the CAM, many different sizes can be obtained.

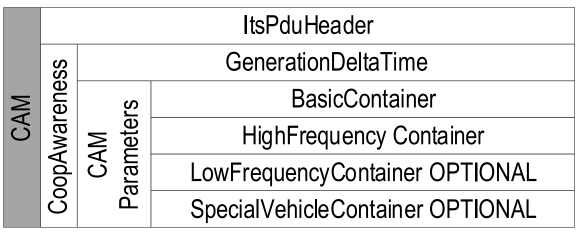


Figure 2: CAM structure defined by ETSI [25].

2.2.1.2 CPM

A CPM includes an ITS PDU header and 4 types of containers: one Management Container, one Station Data Container, one or more Sensor Information Containers, and one or more Perceived Object Containers (POCs) [26] (see Figure 3).

The Management Container is mandatory and provides basic information about the transmitting vehicle, including its type and position. The position is used to reference the detected objects. The Station Data Container is optional and includes additional information about the transmitting vehicle, such as its speed, heading, or acceleration. Part of this information is also included in the CAM transmitted by the same vehicle, but it is also needed in the CPM. If this information were not included in the CPM, the transmitting vehicle dynamics would need to be estimated by the receiving vehicle from the last received CAM. This estimation could reduce the accuracy of the positioning and speed estimation of the transmitting vehicle and its perceived objects.

The Sensor Information Containers describe the sensing capabilities of the transmitting vehicle. The Sensor Information Containers are used by receiving vehicles to derive the areas that are currently sensed by nearby vehicles. A Sensor Information Container includes the ID of a sensor, its type (e.g. radar, lidar or a sensor fusion system) and its detection area, among other Data Elements. Up to ten Sensor Information Containers can be included in a CPM.

The POCs describe the dynamic state and properties of the detected objects. Each POC includes information about a detected object, including its object ID, the ID of the sensor that detected it, the time of measurement, the distance between the detected object and the transmitting vehicle in the XY-plane, and the speed and dimensions of the object, among others. A single CPM can include up to 255 POCs. Multiple POCs could report information about the same detected object but obtained with different sensors. Alternatively, the sensed information could also be fused and reported in a single POC. The first approach reduces the computational needs and processing delays at the transmitting vehicle but may increase the channel load and processing needs at the receiver. The length of the CPM depends on the number of optional containers considered, e.g. number of detected objects and onboard sensors.

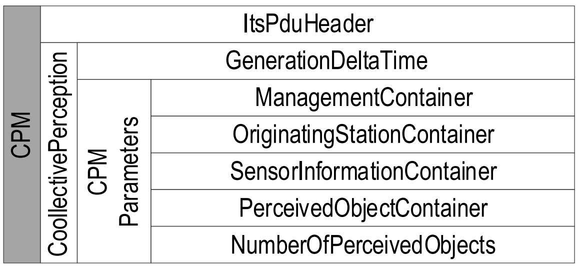


Figure 3: CPM structure defined by ETSI [26].

2.2.1.3 MCM

The MCM includes the ITS PDU header and the containers illustrated in Figure 4. The *GenerationDeltaTime* defines the time at which the MCM has been generated. The *BasicContainer* includes the latest position and the type of the originating station (a vehicle or an RSU). The *ManeuverContainer* can include a *VehicleManeuverContainer* if is transmitted by a vehicle or a *RSUSuggestedManeuverContainer* if it is transmitted by the road infrastructure.

The *VehicleManeuverContainer* is transmitted by vehicles and includes the planned trajectory and optionally the desired trajectory. The container also includes the vehicle dynamics object that includes information such as the heading, speed, acceleration or lane position. Each trajectory contains a variable number of trajectory points, each of them with its coordinates relative to the vehicle position (*deltaXCm* and *deltaYCm*), the remaining time to reach the point (deltaTimeMs), and the vehicle heading and speed when it reaches the point (*headingValue* and *absSpeed*).

The *RSUSuggestedManeuverContainer* includes different data elements so that RSUs can support the coordination of maneuvers. The *intersectionReferenceID* and the *roadSegmentReferenceID* are used as geographical references of the advices or notifications contained in the MCM. Any lane ID employed in the MCM will refer to this specific intersection or segment of the road. This container includes the vehicle advice list composed of a list of vehicle advice objects. Each vehicle advice is sent to a specific vehicle that is identified by the Target Station ID. Four types of advices are possible: lane advice, car following advice (speed and gap advice), ToC advice, and safe spot advice.

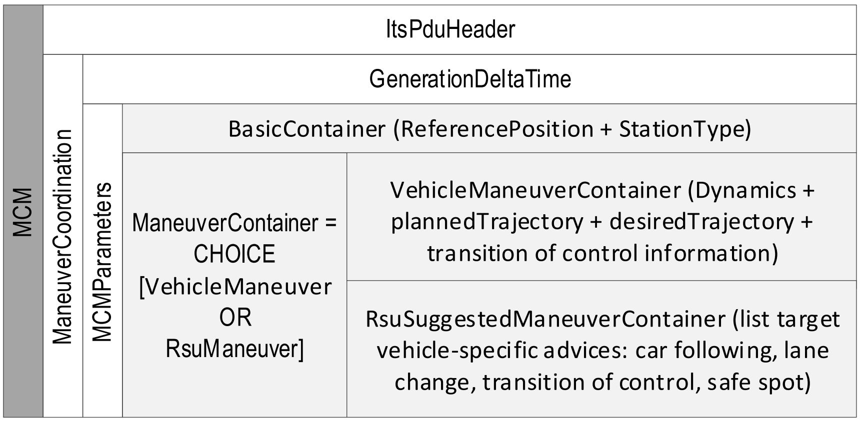
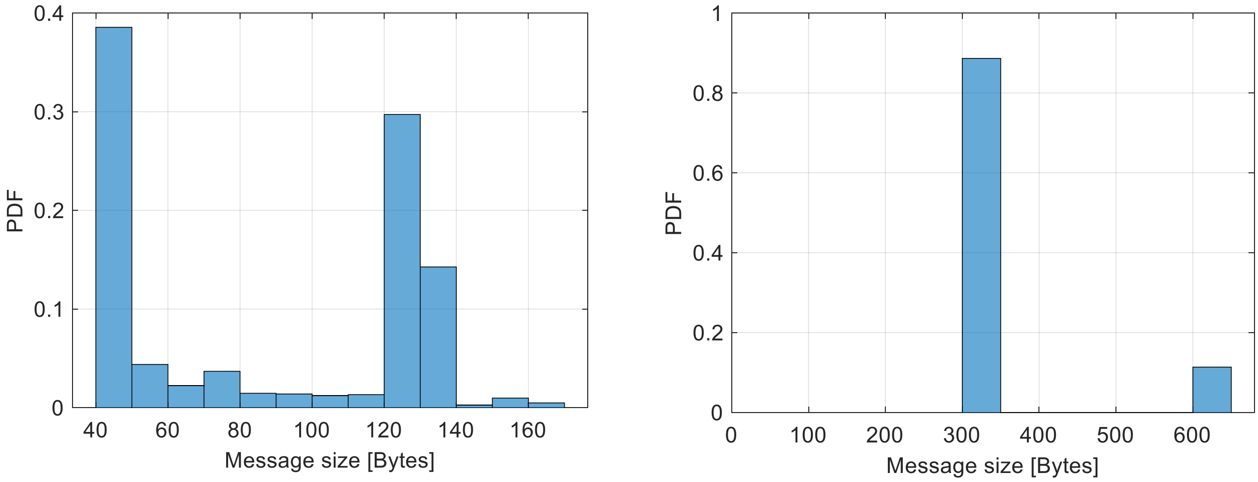


Figure 4: MCM structure defined by TransAID [24].

2.2.2 Compression bounds

To conduct this study, real CAM, CPM and MCM messages obtained in TransAID WP7 [12] have been considered. The use of V2X messages obtained in real-world experiments is important because the properties of the data to be compressed can influence the compression gain that can be achieved. We have therefore used in this study a set of CAMs obtained in real world environments with variable sizes that follow the PDF (Probability Density Function) depicted in Figure 5a. As it can be observed, most of the CAMs used have a message size of around 40 Bytes or 130 Bytes. The CPMs considered in this study were generated and transmitted by an RSU located in an intersection. All the CPMs collected have a size of 105 Bytes. The MCMs used in this study were transmitted by 2 vehicles in a merging highway scenario. All MCMs contain the planned trajectory. When a manoeuvre coordination is required, they also contain the desired trajectory. As a consequence, the MCMs used in this study have 2 different sizes (either 329 Bytes or 608 Bytes), depending on whether they contain the desired trajectory or not. The PDF of the size of the MCM messages used is presented in Figure 5b.



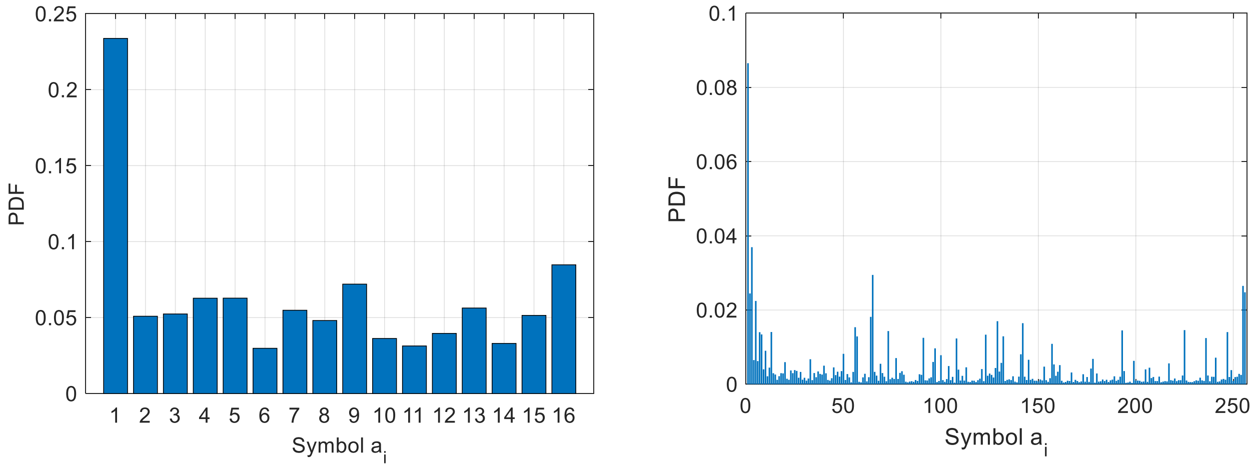
(a) CAM **(b) MCM**

Figure 5: PDF (Probability Density Function) of the sizes of the CAMs and MCMs.

An important aspect that influences the limit of the compression gain that can be achieved is the probability of the different symbols in the V2X messages considered. To estimate this limit, we have analysed each type of V2X message considering 3 different alphabets, each alphabet containing $K=2^4=16$, $K=2^8=256$ and $K=2^{12}=4096$ symbols:

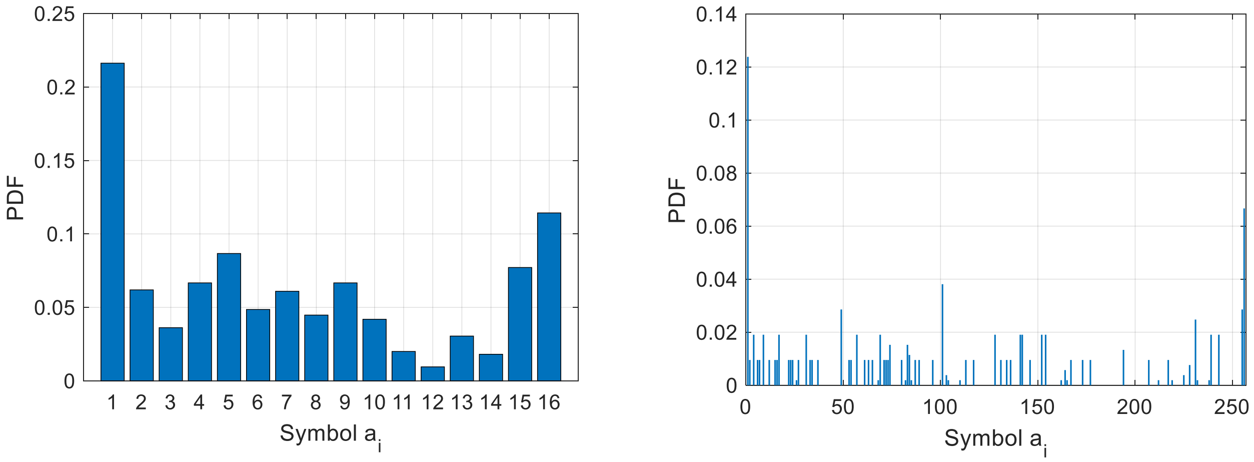
- K=16. Each symbol can be represented by 4 bits without compression. The alphabet is the set of hexadecimal symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.
- K=256. Each symbol can be represented by 8 bits without compression. The alphabet is the set of all pairs of possible hexadecimal symbols: 00, 01, 02, …, FE and FF.
- K=4096. Each symbol can be represented by 12 bits without compression. The alphabet is the set of all groups of 3 of possible hexadecimal symbols: 000, 001, 002, …, FFE and FFF.

We have analysed all the messages used in this study to calculate the probability of each symbol for each type of message and each alphabet. Figure 6, Figure 7 and Figure 8 show the PDF of the symbols of the CAMs, CPMs and MCMs used in this study. The case of K=4096 is not shown because it cannot be clearly plotted due to its high number of symbols. These figures clearly show that some symbols have higher probability than others, which increase the potential of compression algorithms to achieve large compression gains. In fact, when all symbols have the same probability, the entropy is maximized and therefore the limit of the compression gain is minimized.



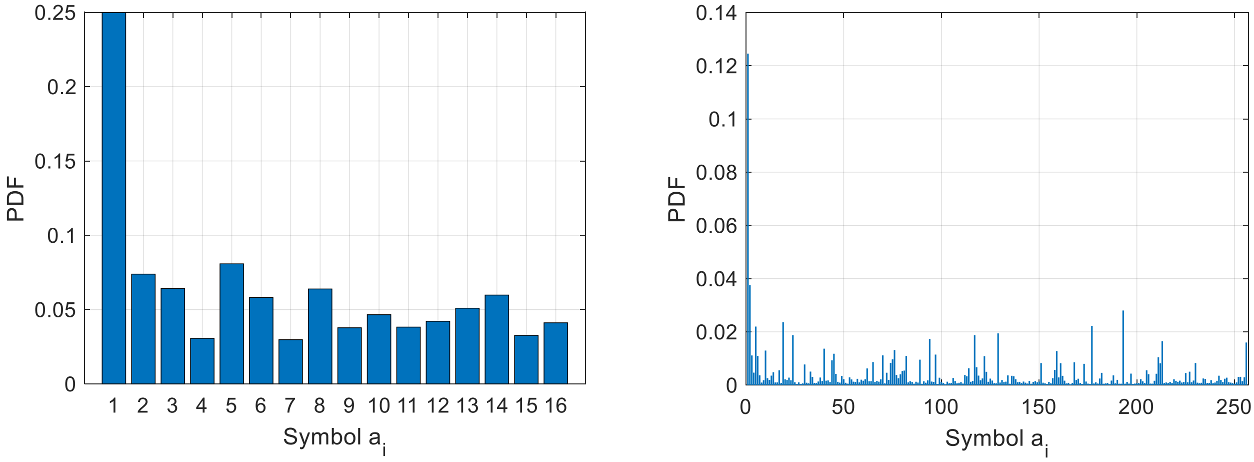
(a) K=16 symbols **(b) K=256 symbols**

Figure 6: PDF (Probability Density Function) of the symbols of a CAM.



(a) K=16 symbols **(b) K=256 symbols**

Figure 7: PDF (Probability Density Function) of the symbols of a CPM.



(a) K=16 symbols **(b) K=256 symbols**

Figure 8: PDF (Probability Density Function) of the symbols of a MCM.

The probabilities of the different symbols have been used to calculate the dictionary of codewords with the Shannon-Fano algorithm. This entropy compression algorithm assigns a higher number of bits to those symbols with lower probability. Table 1 presents the dictionaries of codewords for CAM, CPM and MCM obtained with the Shannon-Fano algorithm for K=16 symbols. Without compression, each symbol would be represented by 4 bits. However, thanks to the compression algorithm the symbols with higher probability (e.g. symbol "0", see Figure 6, Figure 7 and Figure 8) have fewer number of bits. The same procedure has been followed for K=256 and K=4096, but their dictionaries have not been shown in this document for readability reasons.

Table 1: Codewords dictionaries using Shannon-Fano algorithm with *K*=16 symbols

Symbol	CAM	CPM	MCM
0	00	00	00
1	010	011	0110
2	1000	100	0111
3	1010	1010	11110
4	11000	10111	010
5	11011	11001	1010
6	11001	11010	11111
7	1011	1110	1000
8	011	010	1110
9	111110	11000	10111
A	1001	10110	11011
B	11111	111111	1100
C	11010	111110	10110
D	11110	11110	1001
E	11100	11101	11101
F	11101	11011	11010

The sets of symbol probabilities computed for the different message types and alphabets have been used to compute the entropy $H(X)$ of the symbol generation, and the theoretical compression gain that could be achieved with entropy compression algorithms, such as the Shannon-Fano algorithm. This theoretical compression gain is presented in Table 2, together with the entropy values computed. The obtained results show that compression gains up to approximately 50% could be achieved, demonstrating the potential of data compression to reduce the load and improve the V2X reliability. A non-negligible compression gain of 7-8% could be achieved even for K=16. The compression gain increases as the number of symbols in the alphabet increases. Larger alphabets would be possible and would provide higher compression gains, although we stopped at K=4096 symbols. The main disadvantage of having a large alphabet is the need for higher computing capabilities to search and replace each symbol with its corresponding codeword.

Table 2: Entropy and theoretical compression gain

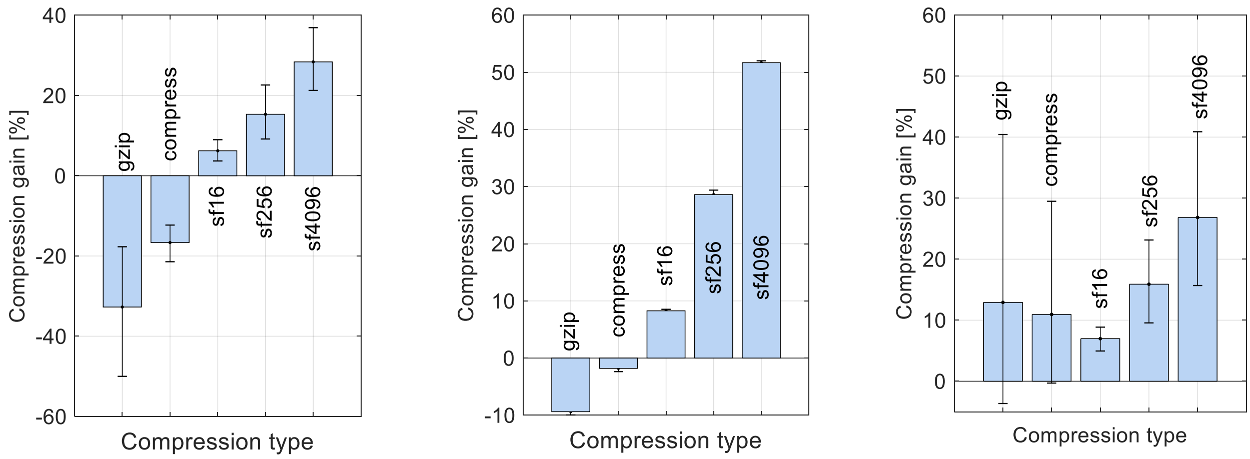
Number of symbols *n*	Entropy *H(X)*			Theoretical compression gain		
	CAM	**CPM**	**MCM**	**CAM**	**CPM**	**MCM**
16	3.73	3.65	3.69	6.7%	8.6%	7.7%
256	6.89	5.66	6.73	13.8%	29.2%	15.9%
4096	8.68	5.73	8.85	27.6%	52.2%	26.2%

2.3 Compression of V2X messages

Five different compression methods have been used in this study to compress the set of real CAMs, CPMs and MCMs messages previously described and compare their performance. These methods include gzip and compress (based on adaptive dictionary compression) and Shannon-Fano (based on entropy compression). The later was evaluated with three alphabet sizes (*K*=16, 256 and 4096) and will be referred to as *sf16*, *sf256* and *sf4096*. Figure 9 shows the compression gain achieved with the five compression methods used, differentiating between CAMs, CPMs and MCMs. While the bars represent the average values obtained, the vertical lines represent the 5^{th} and 95^{th} percentiles.

The first aspect to be highlighted in Figure 9 is the perfect matching between the average values obtained with the Shannon-Fano algorithm and the theoretical bounds calculated and presented in Table 2. The variations with respect to the mean value are produced in V2X messages with a symbol distribution that deviates from the distributions calculated for the whole set of messages and depicted in Figure 6, Figure 7 and Figure 8. The same trends are obtained with the Shannon-Fano algorithm for CAMs, CPMs and MCMs. The highest compression gain is achieved with the largest alphabet (*sf4096*).

Different trends are obtained with adaptive dictionary compression methods (gzip and compress). The results reported in Figure 9a and Figure 9b show that gzip and compress produce a negative compression gain for both CAMs and CPMs, thus increasing the message size after compression. This negative effect is produced because these two messages are composed by multiple containers without any pattern that is significantly repeated and that could be therefore efficiently compressed by this type of compression algorithms. However, gzip and compress are able to obtain high compression gains for MCMs because a large portion of this message is occupied by the planned and desired trajectories. Each trajectory can have 30 trajectory points, and thus can have long sequences of repeated or similar values when e.g. the vehicle is moving at constant speed and heading.



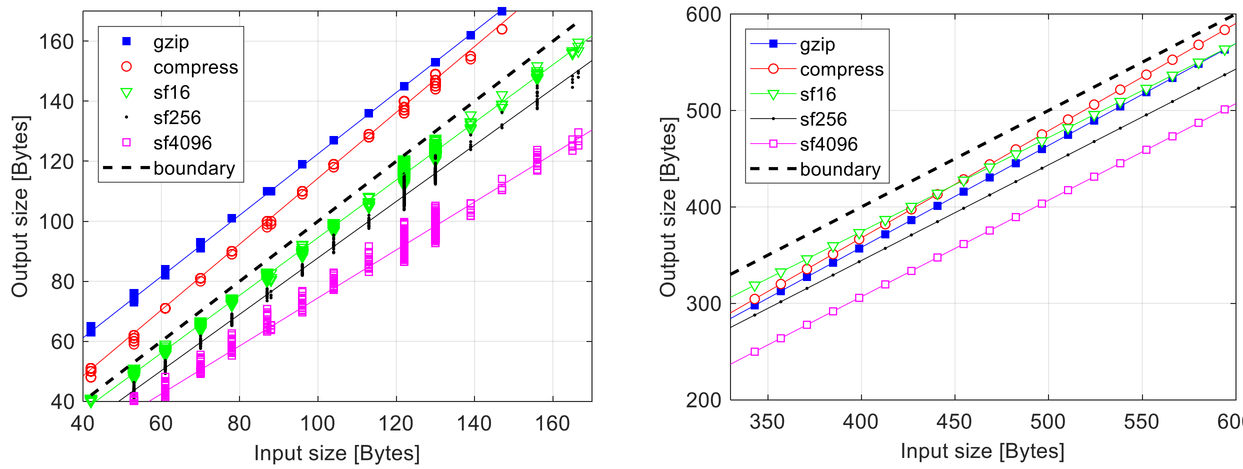
(a) CAM (b) CPM (c) MCM

Figure 9: Compression gain.

To analyze the influence of the message size, Figure 10 depicts the relationship between the output size (S_o) –after compression– and the input size (S_i) –before compression– for CAM and MCM. The markers in Figure 10a (CAMs) show all the input/output values for the five compression methods used. In Figure 10b (MCMs), the markers are just used to differentiate the lines, since we only had 2 different MCM sizes. In both figures, the dashed line represents the compression boundary where S_o=S_i. All the points or curves below this boundary correspond to successful compression methods that could effectively reduce the message size. The solid lines represent the linear models that better fit the obtained results for each compression method and message. These models follow the linear equation:

$$S_o = \alpha \cdot S_i + \beta \tag{6}$$

where α and β are the parameters of the models and their values are presented in Table 3. These models can be used to estimate the compressed size of a message as a function of the input size, without implementing any compression algorithm, and thus can be useful for simulation studies.



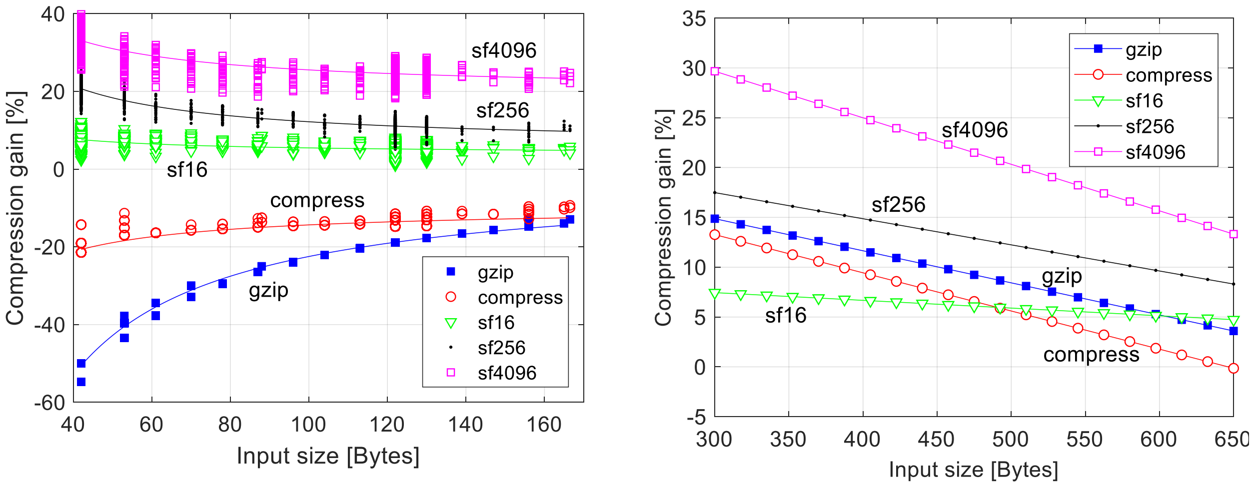
(a) CAM (b) MCM

Figure 10: Compressed size as a function of the input size.

Table 3: Parameters of the analytical models of V2X message compression

Compression type	CAM		CPM		MCM	
	α	β	α	β	α	β
gzip	1.022	20.32	1.0938	0	1.056	-64.4
compress	1.098	4.572	1.0180	0	1.111	-76.66
sf16	0.9607	-1.536	0.9173	0	0.9749	-15.5
sf256	0.9396	-6.158	0.7140	0	0.9921	-52.44
sf4096	0.7993	-5.472	0.4832	0	1.001	-93.41

Using the same input/output values as in Figure 10, Figure 11 depicts the compression gain as a function of the input size. The compression boundary is the horizontal line with 0% compression gain. The solid lines are obtained by computing the compression gain achieved with the linear models previously described. As it can be observed, the compression gain of the Shannon-Fano algorithm tends to decrease as the input size increases, for both CAMs and MCMs. This effect is produced in part because the dictionaries of codewords were optimized for the set of real messages available, that contained a large portion of messages with small size (see Figure 5). Figure 11 also reveals different trends for adaptive dictionary compression methods (gzip and compress) for CAMs. Their compression gain increases as the input size increases, but the relative low size of CAMs does not allow to exploit the potential of this type of methods, and in particular, of gzip. Larger messages could result in positive compression gains using gzip, as shown in Figure 12. The results shown in this figure were obtained by artificially generating CAMs with larger sizes, following the same PDF of symbols than the standardized CAMs (Figure 6).



(a) CAM **(b) MCM**

Figure 11: Compression gain as a function of the message size.

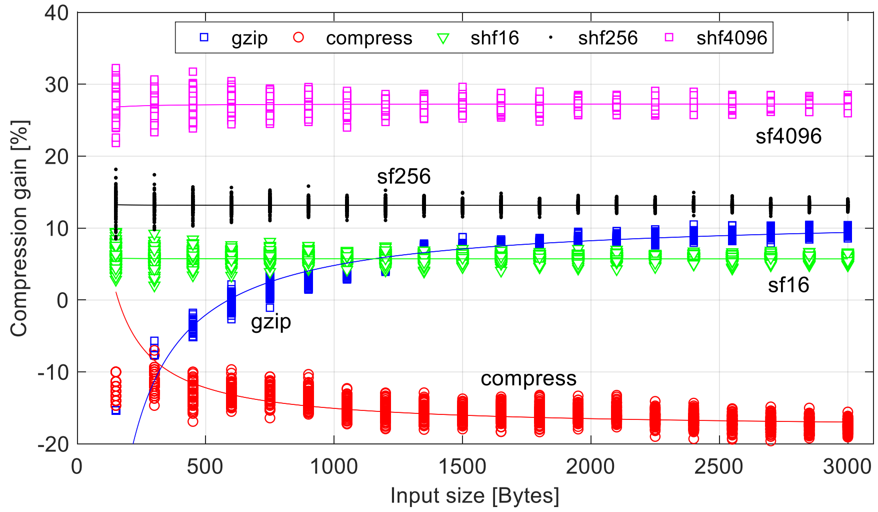


Figure 12: Compression gain as a function of the CAM size for randomly generated CAMs.

2.4 Communications performance

To analyse the impact of V2X message compression algorithms on the reliability of V2X communications, we have conducted a network simulation study using ns-3 (https://www.nsnam.org/). This simulator has also been used in TransAID to evaluate the communications performance of e.g. collective perception and maneuver coordination [41]. All vehicles are assumed to be equipped with an ITS-G5 transceiver (100% penetration rate) and operate in the same channel. Vehicles do not apply any DCC (Decentralized Congestion Control) method to focus on the evaluation of the compression methods. All vehicles will generate and transmit CAMs, CPMs and MCMs. CAMs are generated depending on the vehicle dynamics following the ETSI generation rules (e.g. a vehicle generates a CAM every 4m). The size of each CAM is randomly selected following the PDF of Figure 6. CPMs are also generated following the ETSI basic generation rules, and we have considered that all vehicles are equipped with a 360º sensor. The size of each CPM depends on the number of detected objects and the generation rules. Since the MCM generation rules are still under definition, we have considered that all vehicles transmit MCMs at a fixed rate equal to 5 Hz. The size of each MCM is randomly selected following the PDF of Figure 8.

The traffic scenario is a six-lane highway with 5km length and a lane width of 4 meters. To avoid boundary effects, statistics are only taken from the vehicles located in the 2 km around the centre of the simulation scenario. We simulate three different traffic densities: 180 veh/km, 120 veh/km and 60 veh/km. The configuration of the scenario is summarized in Table 4. The propagation effects are modelled using the Winner+ B1 propagation model following 3GPP guidelines [27]. The communication parameters are summarized in Table 5.

Table 4: Scenario parameters

Parameter	**Traffic density**		
	Low	**High**	**Very high**

<table>
<tr><td>Highway length</td><td colspan="3">5 km</td></tr>
<tr><td>Number of lanes</td><td colspan="3">6 (3 per driving direction)</td></tr>
<tr><td>Traffic density</td><td>60 veh/km</td><td>120 veh/km</td><td>180 veh/km</td></tr>
<tr><td rowspan="3">Speed per lane</td><td>140 km/h</td><td>70 km/h</td><td rowspan="3">50 km/h</td></tr>
<tr><td>132 km/h</td><td>66 km/h</td></tr>
<tr><td>118 km/h</td><td>59 km/h</td></tr>
</table>

Table 5: Communication parameters

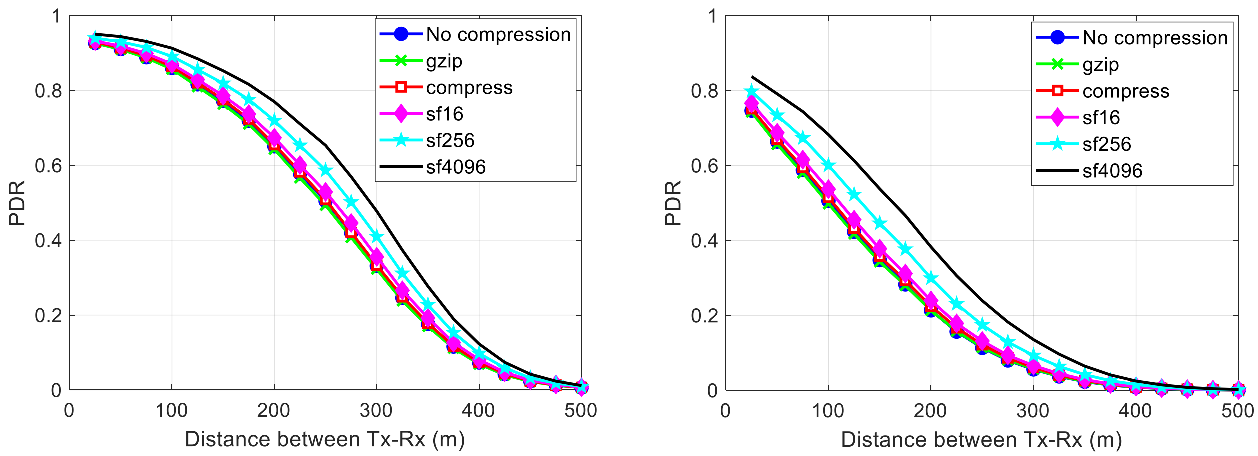
Parameter	Values
Transmission power	23 dBm
Antenna gain (tx and rx)	0 dBi
Channel bandwidth/carrier freq.	10 MHz / 5.9 GHz
Noise figure	9 dB
Energy detection threshold	-85 dBm
Data rate	6 Mbps (QPSK 1⁄2)

The potential of message compression to improve the performance of V2X networks is first analyzed in this study by evaluating the channel load. The channel load is measured through the CBR (Channel Busy Ratio), that is defined as the percentage of time that the radio channel is sensed as busy. Table 6 shows the average CBR obtained for the three traffic densities considered, and all the compression methods evaluated (the relative difference with the scenario without compression is noted between parentheses). The highest reduction of the channel load is achieved with sf4096, in line with the results presented in the previous section. The CBR can be reduced thanks to V2X message compression up to around 27%, which is a non-negligible amount given that data compression does not reduce the amount of information transmitted. It is interesting to note that the average reduction of the CBR is not equal to the average compression gain achieved. For example, the average compression gain with sf4096 was between 27% and 52% approximately (Figure 9), but the CBR is reduced between 18% and 27%. This effect is produced because the compression is applied at the Facilities layer, and the headers added at the Transport & Network, Access and PHY layers are not compressed. The application of the compression methods at e.g. the Access layer could therefore more directly transfer the compression gain to the CBR reduction, and thus produce even better results (if the information added by the lower layers could be effectively compressed). It is also worth noting that the relative reduction of the CBR decreases if the traffic density increases. This effect is related to packet collisions. When the traffic density and the CBR increase, the number of packet collisions also augment. When two or more packets overlap in time, their contribution to the CBR is reduced. When no compression is applied, higher packet collisions are produced, especially for the very high traffic density scenario.

Table 6: Average CBR (Chanel Busy Ratio) experienced

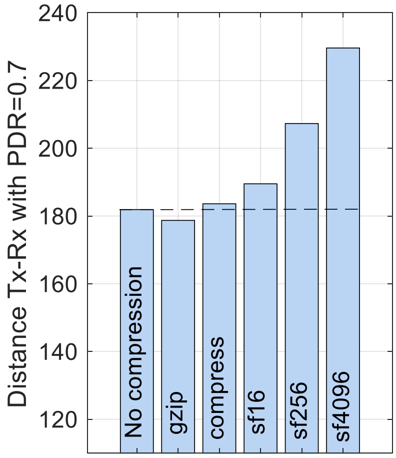
Compression method	Traffic density		
	Low	High	Very high
No compression	41.80%	61.80%	73.31%
gzip	42.54% (+1.8%)	62.08% (+0.5%)	73.43% (+0.2%)
compress	41.45% (-0.8%)	60.98% (-1.3%)	72.46% (-1.2%)
sf16	39.80% (-4.8%)	59.51% (-3.7%)	71.18% (-2.9%)
sf256	35.70% (-14.6%)	54.64% (-11.6%)	66.61% (-9.1%)
sf4096	30.61% (-26.8%)	48.31% (-21.8%)	60.36% (-17.7%)

The reduction of the channel load has a positive effect on the V2X communications reliability. When the channel load decreases, the number of packets lost due to collisions and interferences also decreases. The reliability improvement can be observed through the evaluation of the PDR (Packet Delivery Ratio), defined as the probability of correctly receiving a V2X message at a certain distance to the transmitter. Figure 13 plots the PDR experienced in the low and very high traffic density scenarios for the different compression methods considered. The compression methods with highest compression gains (and thus higher CBR reductions) present the highest PDR increase compared to the scenario without compression. This increase augments the distance at which a connected vehicle can be detected with CAMs, at which a detected object can be detected with CPMs or the distance at which a manoeuvre coordination can take place with MCMs. A common metric to compare the reliability of different solutions is the distance at which a given PDR is achieved. This distance is shown in Figure 14 for PDR=0.7 to more clearly show the gain achieved with data compression. As it can be observed, the use of V2X message compression can significantly increase the distance at which a PDR of 0.7 is achieve. In the very high-density scenario, this distance is nearly doubled with sf4096 compared with the scenario without compression.

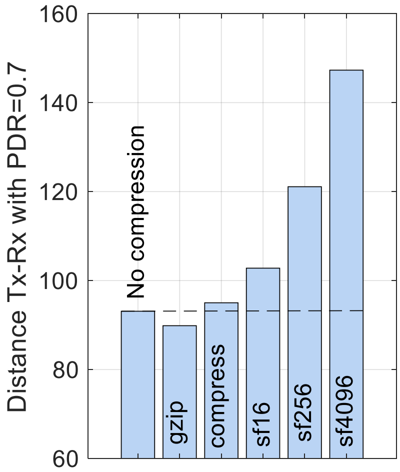


(a) 60 veh/km **(b) 180 veh/km**

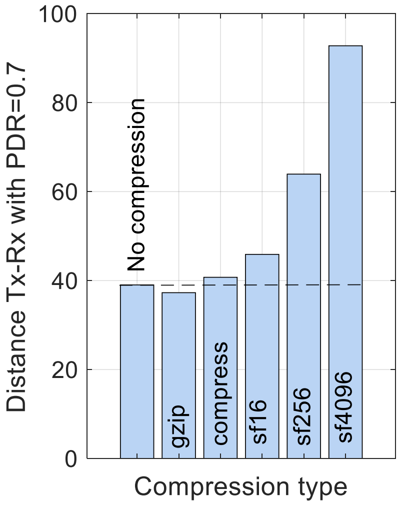
Figure 13: PDR (Packet Delivery Ratio) with and without V2X message compression.



(a) 60 veh/km



(b) 120 veh/km



(c) 180 veh/km

Figure 14: Distance (m) between Tx-Rx at which a PDR of 0.7 is achieved.

3 Congestion control for enhanced V2X reliability

The bandwidth of the wireless communication technologies is a finite resource that needs to be shared among the neighbouring vehicles located in each area. If the number of data transmissions required exceeds the available bandwidth, the radio channel becomes congested. This results in a significant increase of the channel access time and transmission latency, and also an important packet loss probability due to packet collisions or interferences [28]. This is especially the case due to the emergence of connected and automated vehicles and the increasing number of applications and messages that need to be transmitted can overload the radio channel.

To avoid channel congestion and distribute the radio resources among neighbouring vehicles, a congestion control algorithm is needed. One of the most complete solutions to control congestion in vehicular networks is the so-called Decentralized Congestion Control (DCC) defined by ETSI for the ITS-G5 technology. DCC is a cross-layer function that spans over multiple layers of the protocol stack. To this aim, ETSI has defined DCC_ACC, DCC_NET, DCC_FAC and DCC_CROSS components. All of them have been designed to adhere to the upper limits defined in ETSI EN 302 571 [29] in terms of maximum transmission duration and minimum time interval between two consecutive transmissions.

In this section, we have analysed the impact of DCC on the V2X communications reliability, focusing on cooperative sensing. Section 3.2 evaluates the performance and efficiency of the use of ETSI DCC Access. Section 3.3 extends this study by adding two Facilities Layer DCC algorithms to the system: the one in the current ETSI draft, and one proposed by TransAID. All the identified solutions are compared with the best approach proposed in TransAID to intelligently reduce the load and interferences in collective perception and extensively described and evaluated in deliverable 5.2 [41].

3.1 ETSI DCC Framework

The ETSI ITS Communications architecture includes 4 different DCC components distributed over different layers of the protocol stack, as illustrated in Figure 15: DCC_ACC, DCC_NET, DCC_FAC and DCC_CROSS.

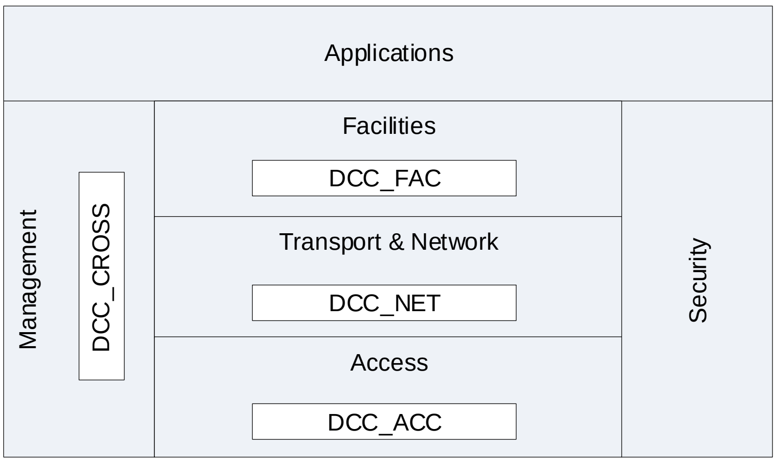


Figure 15: Decentralized Congestion Control (DCC) in the ETSI ITS Communications architecture.

ETSI defines in [30] the DCC_ACC component for ITS-G5. The DCC_ACC component is in the access layer and has been the target of most of the research conducted to date. It operates as a gatekeeper to control the traffic effectively transmitted to the channel. To this aim, the amount of radio resources consumed by the radio transmissions of each vehicle is adapted as a function of the channel load in a decentralized way. The channel load that is used as input for the algorithm can be the one locally measured, or the one provided by DCC_NET if channel load information sharing is supported (see DCC_NET below). DCC_ACC controls the data traffic injected by each vehicle to the radio channel for ITS-G5. It makes use of Prioritization, Queuing and Flow Control, as described below.

Prioritization: The packets that are received by the DCC Access component from the upper layers are first classified according to their priorities. Four different priorities are differentiated, depending on the four traffic classes defined at the Facilities layer: DP0, DP1, DP2 and DP3, where DP0 has the highest priority. At the lower layers, these priorities are mapped to the traffic categories of the ITS-G5 Enhanced Distributed Channel Access (EDCA).

Queuing: DCC Access implements 4 different queues, each of them for one packet priority or traffic class. Each queue follows a first-in-first-out (FIFO) scheduling policy so that the packet that has been waiting longer in the queue is transmitted first. The DCC Access queuing mechanism drops those packets that have been waiting in the queue for a time longer than their lifetime. When a queue is full, no more packets are accepted.

Flow control: Finally, flow control is applied to de-queue packets from the DCC queues and send them to the lower layers for their radio transmission. Packets with higher priorities are de-queued first. A packet is only de-queued if there is no packet with a higher priority waiting in its corresponding queue. As a result, lower priority packets can suffer from starvation and never be transmitted. To control the rate of transmitted packets per vehicle, two approaches have been defined in [30]: Reactive and Adaptive. Both approaches adapt the time between consecutive packet transmissions based on the CBR (Channel Busy Ratio). CBR is defined as the percentage of time that the channel is sensed as busy. The Reactive approach was the only approach standardized in a previous version of the same ETSI standard. Different studies have shown that its performance and stability can be significantly challenged [31][32]. The Adaptive approach was initially proposed in [33] with good stability and convergence properties. Both approaches have been compared in different studies [34], showing the superior performance of Adaptive over Reactive in terms of e.g. packet error ratio or packet inter reception time. These two approaches are described and evaluated in sub-section 3.2.

The DCC_NET component is in the networking & transport layer and defined in [35]. DCC_NET enables the exchange of channel load information (i.e. CBR values) among vehicles so that each vehicle is aware of the channel load experienced by its one-hop and two-hop neighbours. To this aim, each vehicle includes in the networking & transport header in every transmitted single hop broadcast packet the locally measured CBR and the maximum CBR experienced by its one-hop neighbours. The CBR value used by DCC will be the maximum between the CBR values received and the locally measured CBR if DCC_NET is implemented.

The DCC_FAC component is an optional component located in the facilities layer being defined in [36]. DCC_FAC controls the load generated by each application or service taking into account their priorities (or traffic class). To do so, it sub-divides the total amount of resources available for the vehicle to individual resources for each application. If the amount of resources available is not sufficient to transmit the data required by all the applications, the lower priority applications will not be able to transmit. The standardization process of DCC_FAC has not finished yet, due to in part to the inefficiencies identified when integrating it with DCC_ACC [37]. More advanced solutions are being discussed to more intelligently take into account the application needs at the facilities layer to control congestion [38]. We will analyse the existing DCC_FAC component and propose an evolution in sub-section 3.3.

The DCC_CROSS component is located in the transversal management layer and is defined by ETSI in [39]. DCC_CROSS contains for each layer a function that is connected to the interface of the above described DCC components. DCC_CROSS defines the necessary support functions of DCC that needs to be in the management plane and the required interface parameters between the DCC management entity and the DCC entities in the facilities, the networking & transport and the access layers. For example, it computes the available channel resources based on the local CBR and the CBR received by the DCC_NET.

3.2 Access Layer DCC

The DCC_ACC component can be considered the core of the DCC framework defined by ETSI and is therefore the focus of this study. This component allows the implementation of the Reactive or Adaptive approaches.

The Reactive approach is based on a state machine where the current state depends on the CBR and each state can only be reached by a neighbouring state. The Restrictive state is the most stringent one, i.e. the one reached with the highest CBR. The Relaxed state is the least stringent one. Intermediate states called *Active 1*, *Active 2*, … *Active n*, can also be defined. The number of states is not fixed and can be configured. For each state, different radio transmission parameters can be defined to control the channel load depending on the CBR. The ETSI specification allows the adaptation of the data rate or the transmission power, but in related studies only the packet rate (i.e. the time between packet transmissions or T_{off}) is configured differently in each state. In [30], a possible parameter setting is provided as Informative Annex. In this setting, 5 states are defined, and only the packet rate is adapted following the T_{off} values shown in Table 7 for scenarios where the packet duration T_{on} is below 0.5ms. Following this table, if a vehicle requires the transmission of e.g. 8 packets per second and the channel load is 51% (State Active 3), DCC will only allow the transmission of 4 packets per second and will drop the rest. Other configurations are possible, but the one represented in Table 7 is one of the most used in recent studies.

Table 7: Mapping of CBR values to states and T_{off} for T_{on}=0.5ms [30]

State	CBR	Packet rate	T_{off}
Relaxed	< 30%	20 Hz	50 ms
Active 1	30% to 39%	10 Hz	100 ms
Active 2	40% to 49%	5 Hz	200 ms
Active 3	50% to 65%	4 Hz	250 ms
Restrictive	> 65%	1 Hz	1000 ms

The Adaptive approach makes use of a linear control process to adapt the packet rate of each vehicle. This process is designed to make the CBR converge to a target value CBR_{target}=68%. To this aim, it adapts the parameter δ, which is a unitless value that represents the maximum fraction of time that a vehicle is allowed to transmit. The parameter δ is updated every 200 ms based on the difference between the current CBR and the target CBR. Then, the computed δ is used to calculate the time between packet transmissions (T_{off}), taking into account the duration of the current packet

(T_{on}). This approach has been shown to converge to a stable solution in steady state [33]. More details about the standardized parameters and equations can be found in [30].

3.2.1 Evaluation

3.2.1.1 Scenario and settings

In this study we have used the ETSI Collective Perception Service (CPS) [40] as a baseline to analyse and improve its reliability through the use of Access layer DCC. We have therefore followed the ETSI specifications for the CPS. To this aim, we have extended the ns-3 simulator with a CPS component and different on-board sensors.

The configuration of the CPS follows the ETSI specifications and our work in D5.2 [41]. For example, the *T_GenCpm* parameter has been set to 0.1s, so that the maximum CPM rate is 10 Hz. The CPM size is dynamically calculated by the transmitting vehicle based on the number of containers in each CPM. The size of each container has been estimated offline using the current ASN.1 definition of the CPM and the size of the containers that are used in this study is in [41]. In this study, two different sensor configurations, forward and 360º sensors, are analysed. In the first, forward sensors configuration, vehicles are equipped with two forward sensors. The first sensor has a 65m range and a FoV of ±40º. The second sensor has a 150m range and a ±5º FoV. In the second, 360º sensor configuration, vehicles are equipped with a single sensor with 150m range and a 360º FoV. The sensor shadowing effect (sensor masking) is implemented in the XY-plane, and we assume that the sensors can detect only the vehicles that are in their Line-of-Sight (LoS). We assume that the objects detected by the two sensors are fused.

By default, we will consider that vehicles generate CAMs and CPMs based on the ETSI generation rules (see [42] and [40]). To this aim, CAMs are generated depending on the vehicle its own dynamics (e.g. a vehicle generates CAM every 4m), while CPMs are generated depending on the number of detected vehicles (i.e. traffic density) and on their dynamics (e.g. an object is included in a CPM every 4m). The highway scenarios are configured with 5km length and a lane width of 4 meters. We simulate two different traffic densities following the 3GPP guidelines for V2X simulations [43]. The 3 lanes very high traffic density scenario with (180 veh/km) and 4 lanes very high traffic density scenario with (240 veh/km). In both traffic densities, vehicles travel at a maximum speed of 50 km/h (due to the high density). The speeds have been selected based on statistics of a typical US highway obtained from the PeMS database [44]. We analysed the lanes speed of the highway for every hour within a single day and took the average speed for each individual lane. Vehicles created in the simulations have the dimension of 4.8m x 1.8m. To avoid boundary effects, statistics are only taken from the vehicles located in the 2km around the centre of the simulation scenario. The configuration of the scenario is summarized in Table 8. All vehicles are assumed to be equipped with an ITS-G5 transceiver (100% penetration) and operate in the same channel. The propagation effects are modelled using the Winner+ B1 propagation model following 3GPP guidelines [43]. The communication parameters are summarized in Table 9.

Table 8: Scenario parameters

Parameter	Values	
	3 Lanes	4 Lanes
Highway length	5 km	
Number of lanes	6 (3 per driving direction)	8 (4 per driving direction)
Traffic density	180 veh/km	240 veh/km
Speed per lane	50 km/h	

Table 9: Communication parameters

Parameter	Values
Transmission power	23 dBm
Antenna gain (tx and rx)	0 dBi
Channel bandwidth/carrier freq.	10 MHz / 5.9 GHz
Noise figure	9 dB
Energy detection threshold	-85 dBm
Data rate	6 Mbps (QPSK 1⁄2)

With respect to DCC configurations, both Reactive and Adaptive approaches are analysed with different queue lengths and message priorities. For the analysis, a queue length of 2 (Q2) and a queue length of 10 (Q10) are considered. In the same priority configuration (referred as S(ame)), both CAMs and CPMs are configured with the same DCC profile DP2. In the different priority configuration (referred as D(ifferent)), CAMs are configured with the DCC profile DP2 while the CPMs are configured with the lower DCC profile DP3. The combinations of queue length and the message priorities for DCC are summarized in Table 10.

Table 10: DCC Queue Length and Priority Configurations

Configuration	Queue length	CAM DCC Profile	CPM DCC Profile
Q2-D	2	DP2	DP3
Q2-S	2	DP2	DP2
Q10-D	10	DP2	DP3
Q10-S	10	DP2	DP2

3.2.1.2 Results

The results obtained to analyse the reliability that can be achieved thanks to the use of DCC for cooperative sensing are organized in different subsections.

3.2.1.2.1 Operation

Before analysing the performance and efficiency of each DCC configuration, it is necessary to better understand the operation of cooperative sensing. Table 11 shows the average CAM and CPM generation rate for the scenarios with 3 lanes (3L) and 4 lanes (4L) for a scenario without DCC. Vehicles travel at an average speed of 50 km/hr (13.8 m/s) in both scenarios. Thus, the vehicles satisfy the 4m rule [42] for every 300 ms, generating CAMs at the rate of 3.3 Hz. For CPMs, at every time interval, the vehicles detect a new object or any of the previously detected objects have satisfied the CPM generation rules [40] generating a CPM at the rate of 9.4 Hz to 9.6 Hz. When DCC is not incorporated, the messages generated at application level will follow the protocol stack and finally get transmitted at the radio level without any restrictions. So, the generation rate shown in Table 11 signifies the transmission rate for the No-DCC configuration.

Table 11: Average CAM and CPM transmission rate without DCC

Scenario	CAM	CPM
3L	3.3 Hz	9.4 Hz
4L	3.3 Hz	9.6 Hz

When DCC Access is enabled, the DCC_ACC component evaluates the channel load and decides whether to delay, drop each packet or pass it to the radio level for transmission. Table 12 shows the average transmission rate of CAMs and CPMs for different traffic densities and DCC configurations. Comparing the results in Table 11 and Table 12, we can observe that only the CAMs can be fully transmitted (rate of 3.3 Hz) when both CAM and CPM have different priorities (-D configurations). In this scenario, the CPMs are configured with the lower priority queue DP3 and CAMs in queue DP2. In DCC, the lower priority messages are considered for transmission only when the higher priority queue is empty. As a result, when they have different priorities, DCC only drops CAMs when there is no CPM in the queue, and this rarely happens in the considered scenario. When considering the same priority for CAMs and CPMs (-S configurations), the number of CAMs and CPMs transmitted when DCC is applied is significantly lower than those without DCC (Table 11). Table 12 also shows that the queue length does not have a significant influence on the transmission rate and therefore on the number of messages dropped by DCC. When comparing the Reactive and Adaptive approaches, the first one produces a lower message transmission rate, because it starts dropping messages at lower channel loads.

Table 12: Average CAM and CPM transmission rate with DCC

Scenario	Configuration	CAM		CPM	
		Reactive	Adaptive	Reactive	Adaptive
3L	Q2-D	3.3 Hz	3.3 Hz	5.1 Hz	7.2 Hz
	Q2-S	2.7 Hz	3.0 Hz	6.3 Hz	7.4 Hz
	Q10-D	3.3 Hz	3.3 Hz	5.5 Hz	7.1 Hz
	Q10-S	3.0 Hz	2.7 Hz	7.8 Hz	8.0 Hz
4L	Q2-D	3.3 Hz	3.3 Hz	3.9 Hz	4.0 Hz
	Q2-S	2.7 Hz	2.1 Hz	5.6 Hz	6.0 Hz
	Q10-D	3.3 Hz	3.3 Hz	4.1 Hz	4.1 Hz
	Q10-S	2.9 Hz	2.1 Hz	6.1 Hz	6.0 Hz

To analyse in more detail the impact of DCC on the CPM message transmissions, Figure 16 represents the PDF (Probability Density Function) of the number of CPMs transmitted per second and per vehicle for different DCC configurations. Figure 16a and Figure 16c show that the Reactive approach transmits less CPMs than the Adaptive approach (aligned with Table 12). This is because, Reactive approach starts imposing restrictions on the packets with lower channel load as shown in Table 7. On the contrary, the Adaptive approach imposes restrictions on the generated messages only when the channel load is high. Figure 16 and Table 12 also shows that the 4L scenario allows a lower transmission rate than the 3L for both Reactive and Adaptive approaches. This is due to the fact that the channel load is higher in the 4L scenario as more vehicles generate messages in the network and DCC reduces the number of messages that each vehicle can transmit to keep the channel load under control. It is also observed from the figure that when CAMs and CPMs have different priorities (red and pink colour), the CPM transmission rate is lower than when they have the same priority (green and cyan colour) for both Reactive and Adaptive approaches (see also Table 12).

Figure 17 is analogous to Figure 16 but considering CAMs instead of CPMs. The results obtained show that all generated CAMs are transmitted in the network without the DCC configuration (blue colour). When enabling DCC, the CAM transmission rates can be reduced depending on its configured priority. In the different priority configuration, the CAMs are configured with the higher priority queue (DP2) and CPMs are configured with (DP3). In this regard, all the generated CAMs are transmitted in the network (see also Table 12). Whereas in the same priority configuration, both CAMs and CPMs are configured to the same priority queue (DP2). As both messages are configured to the same queue, the CAM transmission rate is reduced when compared with the different priority configuration. When comparing between traffic densities, the same priority configuration has a lower CAM transmission rate in the 4L scenario than the 3L scenario which is due to the same effect reported in Figure 17 (see also Table 12).

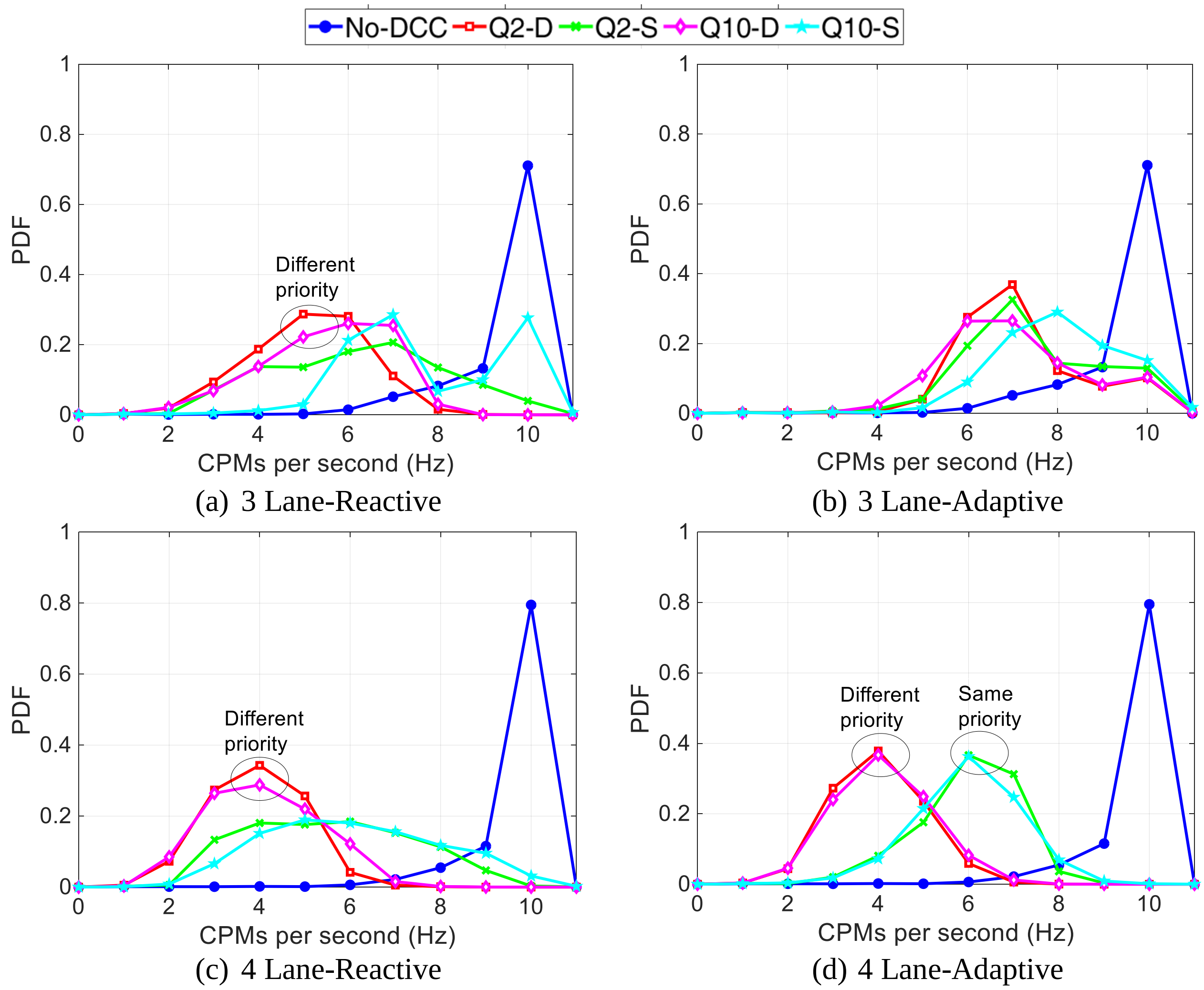


Figure 16: PDF (Probability Density Function) of the number of CPMs transmitted per second and per vehicle.

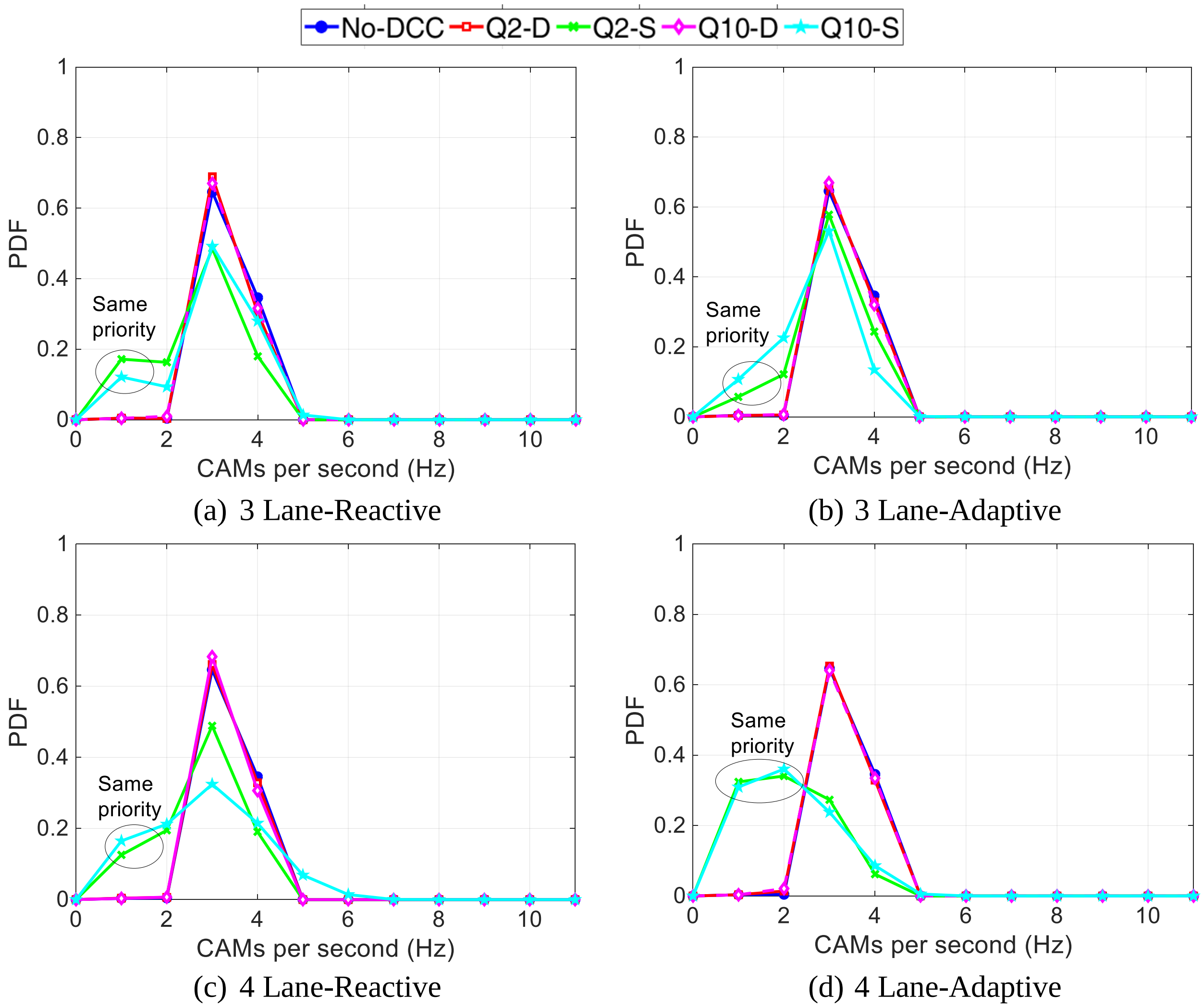


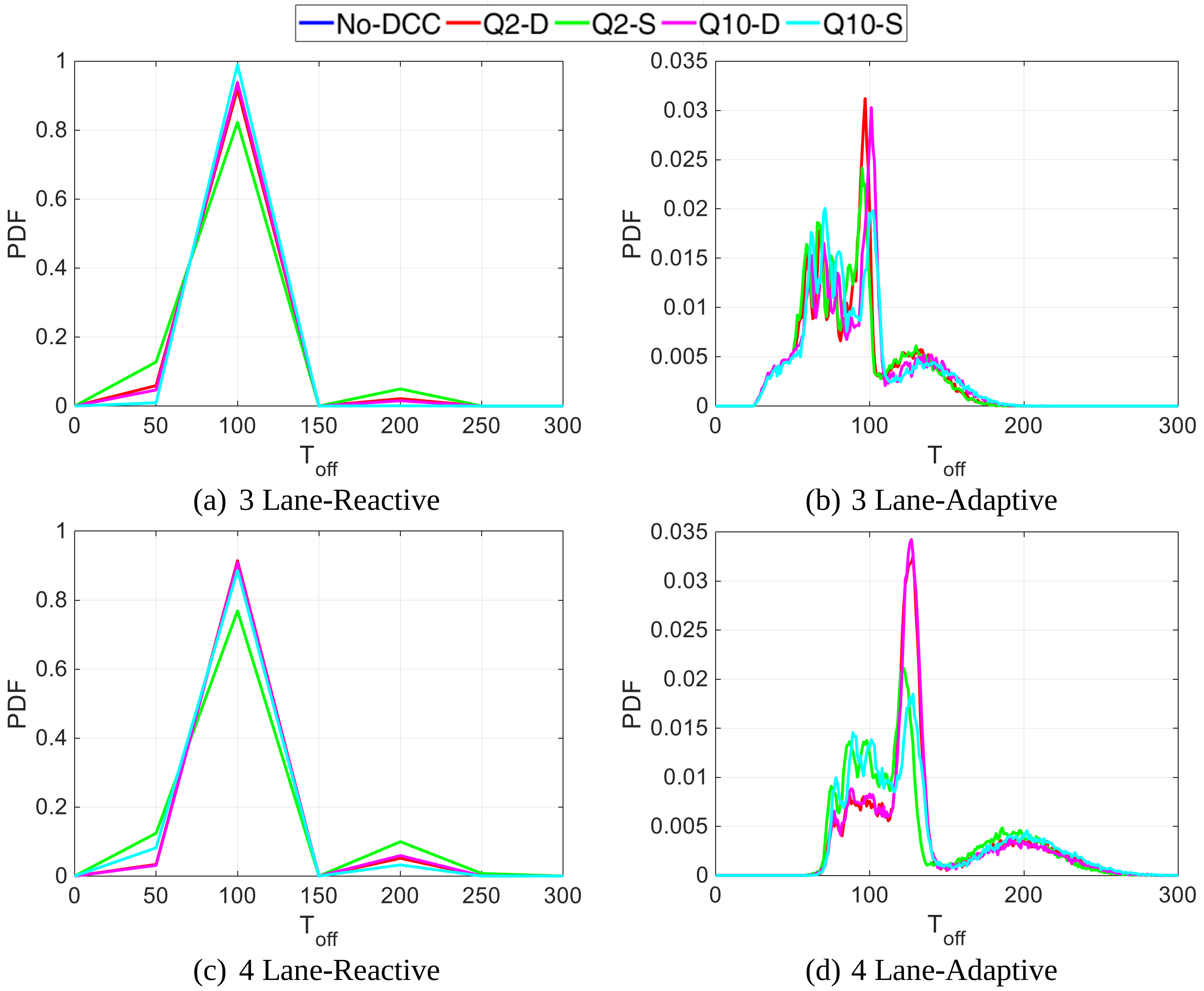
Figure 17: PDF (Probability Density Function) of the number of CAMs transmitted per second and per vehicle.

Table 13 shows the average percentage of CAMs and CPMs dropped per vehicle by the DCC for different scenarios and DCC configurations. DCC drops packets in an uncontrolled fashion based on the channel load without looking into the content of the packet. From the table, the Reactive approach is observed to have an overall higher drop in percentages than the Adaptive approach. This is because the Reactive approach starts dropping packets at lower channel loads (Table 7) which results in a lower transmission rate (Table 12). With higher traffic density, the percentage of packets dropped increases for all the DCC configurations, because of the increase in the number of messages generated in the network. This eventually increases the channel load that prompts DCC to drop more packets (CAMs and CPMs) to limit the channel load. When CAMs and CPMs are configured with different priorities, DCC does not drop any CAMs and all the dropped packets are from the lower priority CPMs. When CAMs and CPMs are configured with the same priority, both CAMs and CPMs are dropped, but the CAMs packet percentage drop is relatively higher than the CPMs percentage drop which is due to the lower CAM generation rate.

Table 13: Percentage of messages dropped due to DCC

Scenario	Configuration	CAM		CPM		CAM+CPM	
		Reactive	Adaptive	Reactive	Adaptive	Reactive	Adaptive
3L	Q2-D	0%	0%	45.5%	23.3%	33.8%	17.4%
	Q2-S	44.3%	17.0%	33.1%	20.7%	35.5%	18.1%
	Q10-D	0%	0%	49.3%	26.2%	36.6%	19.5%
	Q10-S	53.8%	23.3%	19.5%	11.8%	24.3%	14.5%
4L	Q2-D	0%	0%	59.3%	58.0%	44.3%	43.3%
	Q2-S	50.2%	49.8%	42.3%	38.2%	44.0%	40.9%
	Q10-D	0%	0%	62.3%	62.7%	48.7%	46.9%
	Q10-S	54.7%	58.9%	42.8%	41.4%	45.8%	45.7%

The reported dropped packets by the DCC is the direct consequence of the estimated CBR and the computation of the time between packet transmissions or T_{off} which is used to limit the packet rate at the DCC_ACC component. Figure 18 shows the PDF of the T_{off} per vehicle for Reactive and Adaptive approaches under different DCC configurations. It is to be noted that the Y-limits for the different sub-figures are different for better readability. The results shown in Figure 18 demonstrate that the Reactive approach limits the packet transmission rate to 10 Hz (T_{off} =100 ms) most of the time, especially with lower traffic densities (3L scenario). With higher traffic density, the probability of having a T_{off} =200 ms increases for the Reactive approach, limiting the packet transmission rate to 5 Hz. Unlike Reactive, the Adaptive approach makes use of a linear control process to adapt the packet transmission rate. The Adaptive approach estimates the current T_{off} close to the previous estimated T_{off} and reduces the rapid oscillation in the transmission rate. Figure 18b and d show that the Adaptive approach has higher granularity and the T_{off} ranges between 20 ms and 180 ms for the 3L scenario and between 60 ms and 250 ms for the 4L scenario.



(a) 3 Lane-Reactive (b) 3 Lane-Adaptive

(c) 4 Lane-Reactive (d) 4 Lane-Adaptive

Figure 18: PDF (Probability Density Function) of the T_{off} per vehicle.

3.2.1.2.2 Communications performance

This section evaluates the impact of DCC on the communications performance. Table 14 shows the average Channel Busy Ratio (CBR) for different DCC configurations. Without DCC, the CBR increases with the density and hence increasing the risk of saturating the channel. As shown in the table, DCC controls the CBR by dropping packets and hence reduces that risk. In particular, the Reactive approach reduces the CBR significantly by dropping higher number of packets. This is because, the Reactive approach starts dropping packets from 30% of CBR (Table 7) and increases its dropping rate as the CBR increases. A higher CBR is reported for the Adaptive approach. With higher traffic density, the Adaptive approach still maintains the CBR close to the achieved CBR by means of increasing the packet drops since it is designed to converge to a target value CBR_{target}=68%. As it can be observed in the table, the queue length and packet priorities do not have a significant impact on the CBR levels observed.

Table 14: Average CBR (Channel Busy Ratio)

Scenario	Configuration	DCC		No-DCC
		Reactive	Adaptive	
3L	Q2-D	36.2%	61.4%	62.8%
	Q2-S	36.3%	61.3%	
	Q10-D	36.1%	61.8%	
	Q10-S	35.2%	61.9%	
4L	Q2-D	36.9%	62.1%	75.0%
	Q2-S	37.8%	61.9%	
	Q10-D	36.8%	62.1%	
	Q10-S	36.5%	62.1%	

The use of DCC can improve the communications performance because packet dropping reduces the number of packets transmitted over the radio channel, and therefore the CBR and the interference levels generated. This assumption can be considered true if we measure the performance at the radio level, because the probability of correctly receiving a packet that has been effectively transmitted increases. However, from the application perspective, packets dropped by DCC are not transmitted and therefore lost (i.e. not received by any nearby vehicle). As a result, it is important to analyse the impact of DCC on the performance at both the application and radio levels. The communications performance has been quantified in this study through the PDR (Packet Delivery Ratio) metric. At the radio level (referred as Radio), the PDR is defined as the probability of correctly receiving a message that has been effectively transmitted over the radio channel. It has been computed as the ratio between the received and transmitted packets. The PDR at the radio level is therefore influenced by the propagation and interference (i.e. packet collisions) effects. At the application level (referred as App), the PDR is defined as the probability of correctly receiving a message generated by the Facilities layer. It is hence calculated as the ratio between received and generated packets. The PDR at the application level is therefore influenced by the packets dropped by DCC, in addition to the propagation and interference effects. When a packet is dropped by DCC, it is not effectively transmitted, and this represents a packet loss from the application perspective. When DCC is not applied, the PDR at the radio and application level is the same because there is no packet dropped by DCC and all messages generated are transmitted.

Figure 19 shows the PDR for both CAM and CPM packets at the radio and application levels under different DCC configurations for the 3L scenario. Figure 19b and Figure 19d show that the Adaptive approach can improve the PDR at the radio level compared to the No-DCC configuration, thanks to the reduction of the interference and packet collisions. However, the PDR at the application level is significantly lower than the one measured at the radio level. The PDR at the application level of the Adaptive approach is even lower than the PDR of the No-DCC configuration at short distances. Figure 19a and Figure 19c show that the PDR obtained by the Reactive approach at the radio level is significantly lower than the one obtained with the No-DCC configuration. This negative effect is produced because vehicles using the Reactive approach tend to be synchronized with each other at short distances. As a result, they change their T_{off} simultaneously and transmit their packets at the same time, generating packet losses due to collisions. As a result, even though the Reactive approach reduces the interference generated through packet dropping, the PDR is degraded due to a significant amount of packet losses due to collisions. This effect was already identified in the literature [49] and is also observed in our simulation study. At the application level, the PDR obtained with the Reactive approach is significantly lower than the PDR achieved with the No-DCC configuration. The reduction of the channel load due to packets dropping does not compensate the negative effect of not transmitting certain packets at the application level for the Reactive approach. Figure 19 also shows that there is no significant impact of the DCC queue length or the priority of the messages. With the Reactive approach, the use of a shorter queue improves the PDR at the radio level due to the higher number of packets dropped. Also, with the Reactive approach, when both CAM and CPM have the same priority, a slightly higher PDR can be achieved, especially for a queue length of 2.

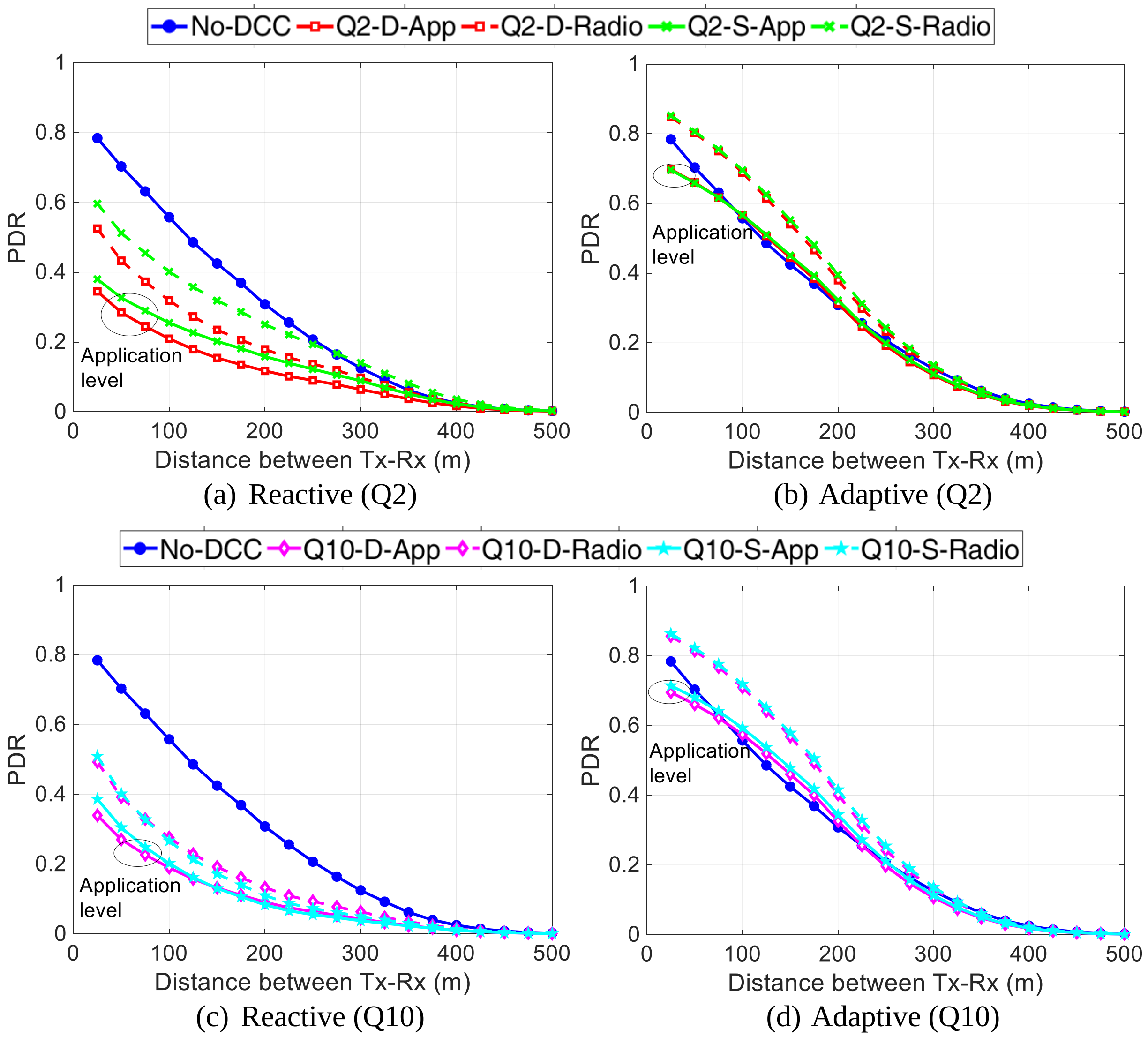


Figure 19: PDR (Packet Delivery Ratio) for CAM+CPM as a function of the distance between transmitter and receiver with the 3L highway scenario.

Figure 20 is analogous to Figure 19, but considers the 4L scenario. The increase of the traffic density produces a general degradation of the PDR due to the higher channel load and packet loses due to either packet collisions or packet dropping. The results obtained for the Reactive approach (Figure 20a and c) follow the same trend as for the 3L scenario, and the PDR curves achieved at the application and radio levels are lower than those achieved for the No-DCC scenario. The results obtained for the Adaptive approach (Figure 20b and d) show a significant improvement of the PDR at the radio level compared to the No-DCC configuration. This is mainly produced due to the reduction of the interference levels thanks to DCC. However, the PDR at the application level is degraded because a significant portion of the messages are dropped by DCC. In this case, the degradation is significant at short distances between transmitter and receiver, compared with the No-DCC configuration.

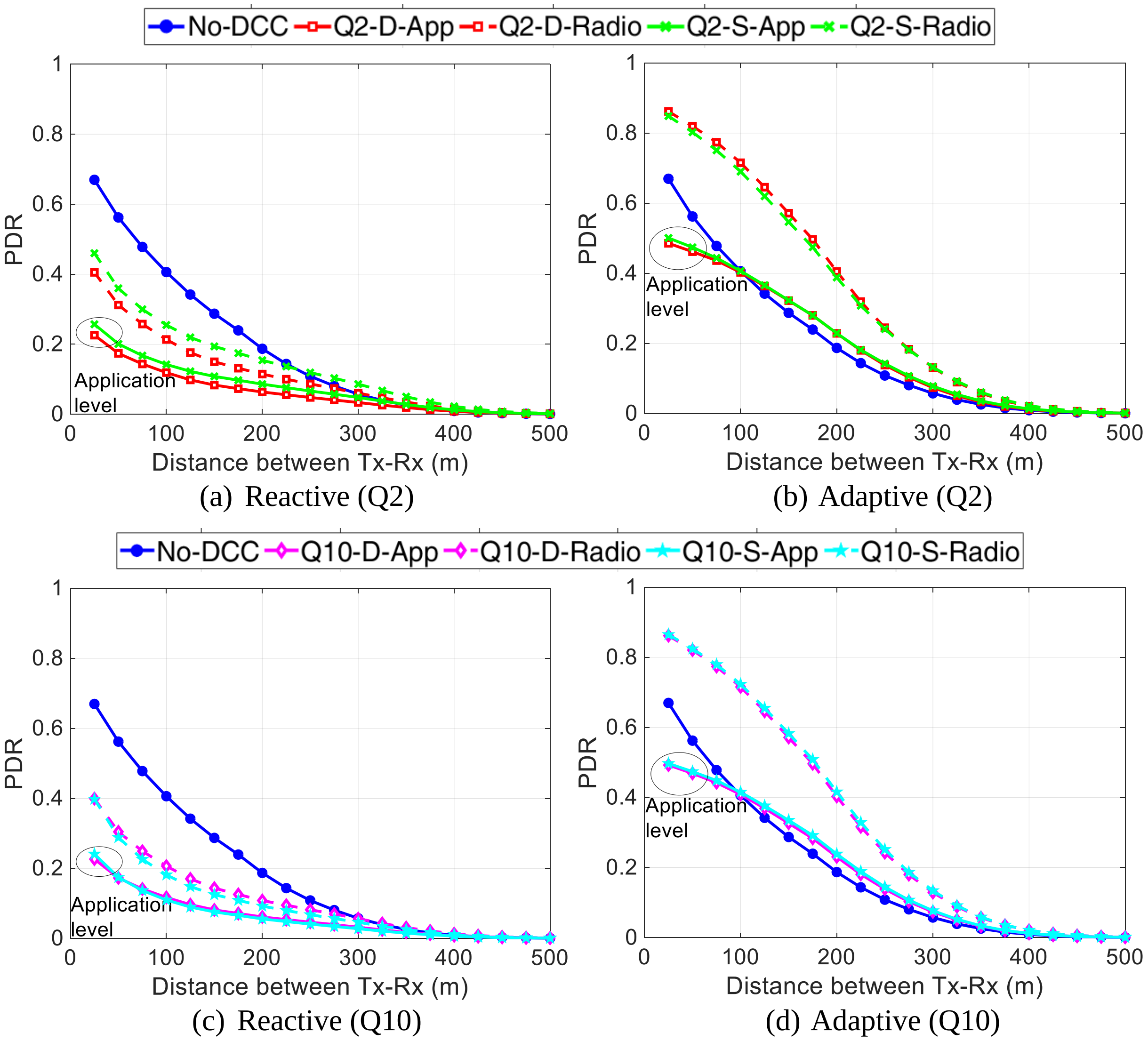
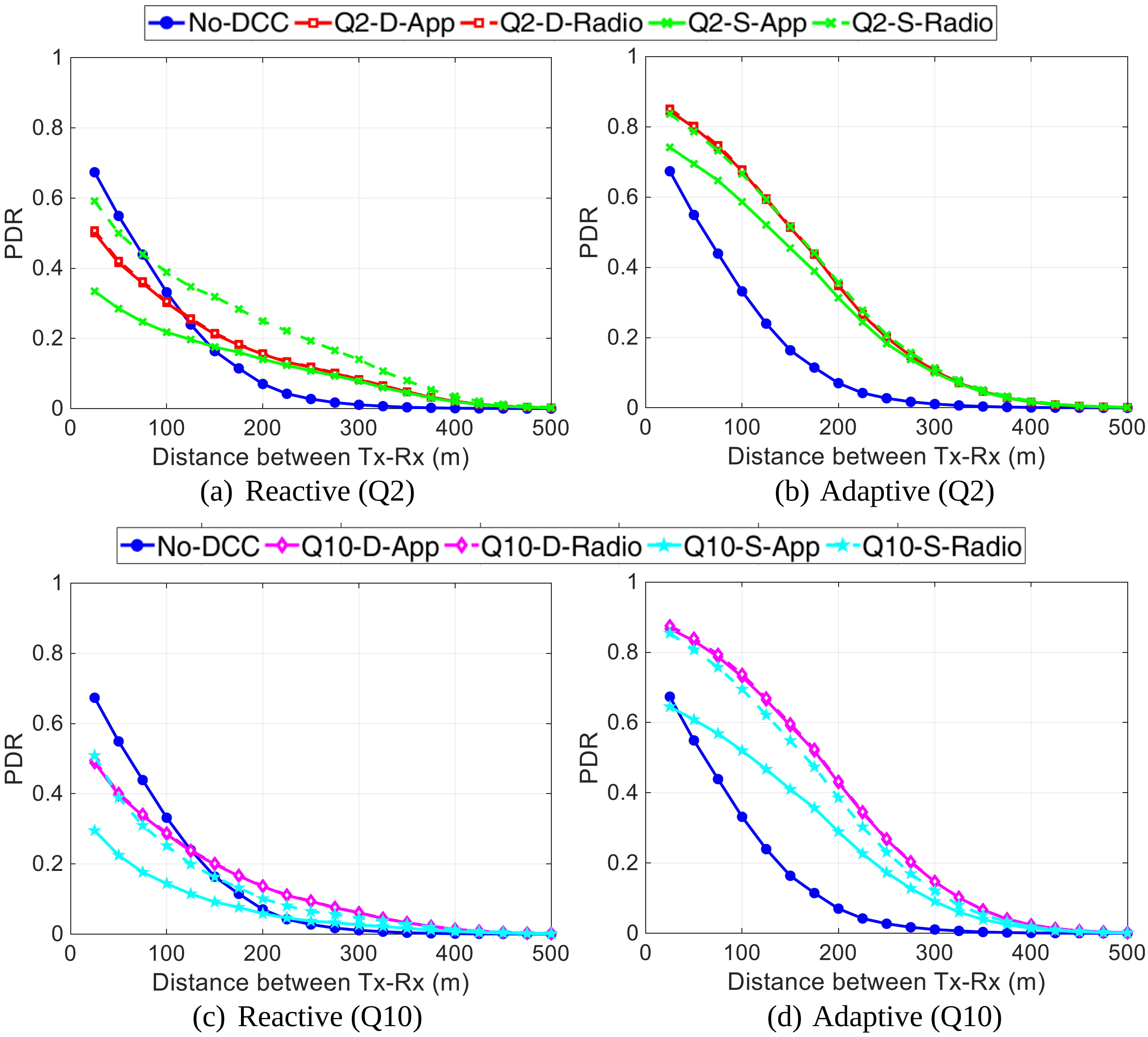


Figure 20: PDR (Packet Delivery Ratio) for CAM+CPM as a function of the distance between transmitter and receiver with the 4L highway scenario.

The previous figures show the PDR experienced when jointly considering CAMs and CPMs. Given the different impact of DCC on CAMs and CPMs depending on their priorities, it is important to analyse them separately as well. Figure 21 shows the PDR for CAM packets at the radio and application levels under different DCC configurations for the 3L scenario. When CAMs and CPMs are configured with different priorities, only CPMs are dropped due to their lower priority. As a result, the PDR obtained for CAMs at the radio and application levels is the same, as it can be observed in Figure 21 (red and pink curves). When both CAMs and CPMs have the same priority, a significant portion of CAMs are dropped and therefore the PDR at the application level is lower than the PDR at the radio level irrespective of the queue length and the DCC approach. With the Reactive approach (Figure 21a and c), the PDR achieved at both the radio and application levels is lower than the PDR achieved without DCC at short distances, demonstrating again the negative effect of DCC in some cases. The use of a short queue can again benefit the PDR, especially at the radio level. With the Adaptive approach (Figure 21b and d), the PDR obtained are significantly higher. The Adaptive approach can better adapt to the channel load conditions and more efficiently drop the necessary packets. As a result, the PDR achieved for CAMs at both application and radio levels is higher than the PDR achieved without DCC.

Similar trends and conclusions are obtained when analysing the PDR for the CAMs in the 4L scenario (Figure 22), although with lower values in general. The PDR without DCC is especially degraded due to the higher interference levels compared to the 3L scenario (Figure 21). However, it is interesting to highlight that both Reactive and Adaptive approaches show a significant degradation of the PDR at the application level due to the higher number of packets dropped, which results in a PDR lower than the PDR without DCC at short distances.



(a) Reactive (Q2) (b) Adaptive (Q2)

(c) Reactive (Q10) (d) Adaptive (Q10)

Figure 21: PDR (Packet Delivery Ratio) for CAM as a function of the distance between transmitter and receiver with the 3L highway scenario.

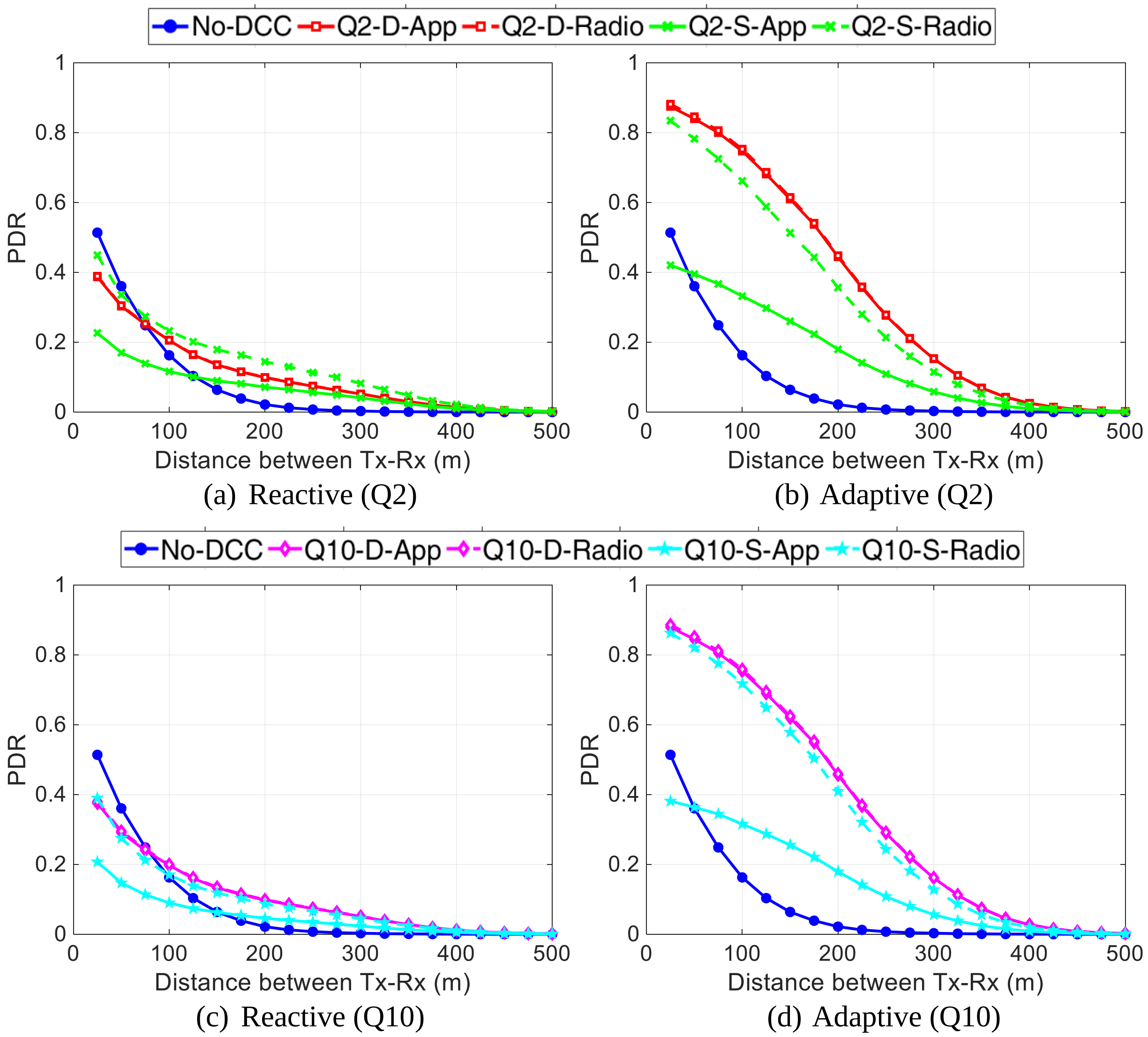
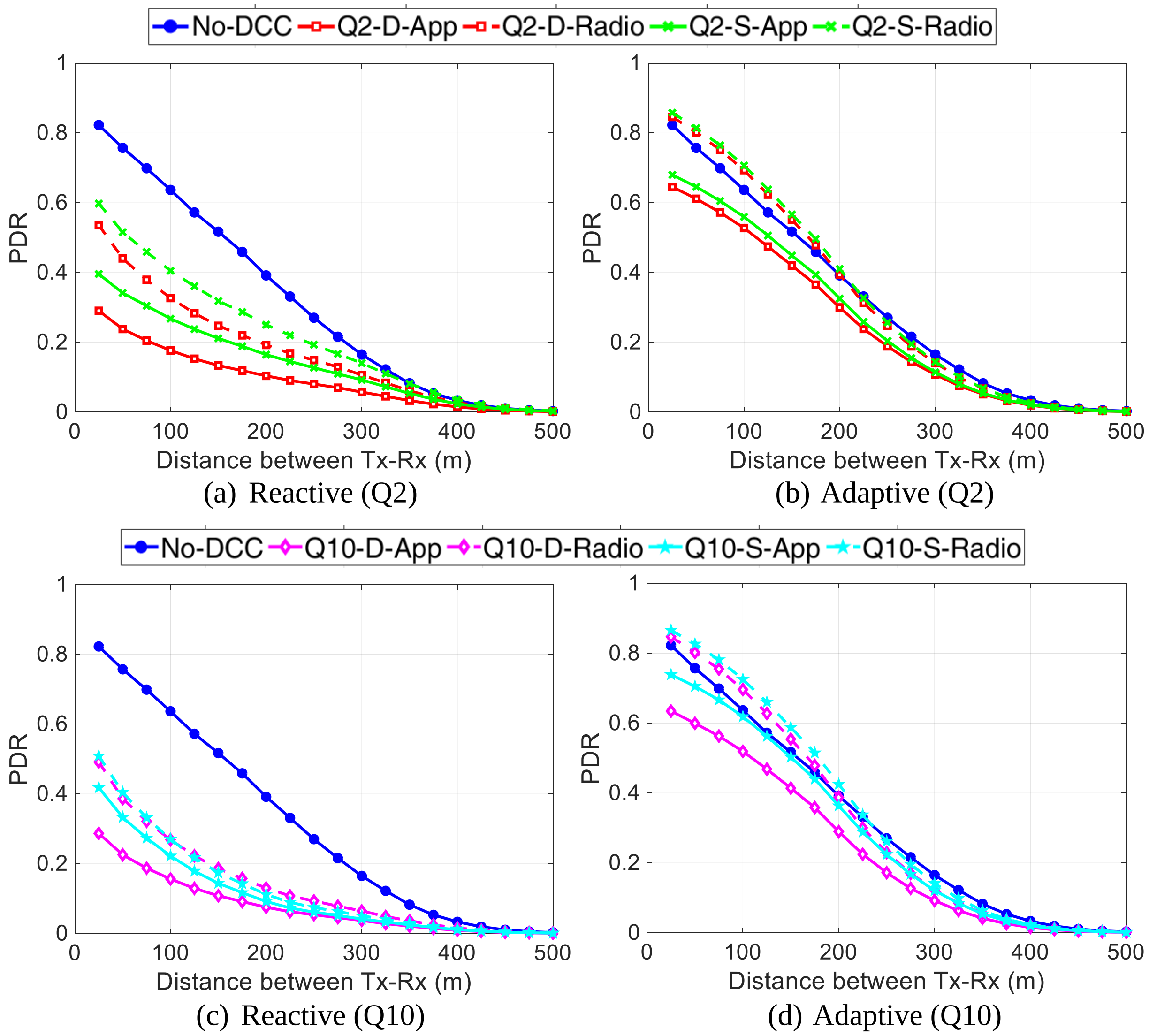


Figure 22: PDR (Packet Delivery Ratio) for CAM as a function of the distance between transmitter and receiver with the 4L highway scenario.

Figure 23 and Figure 24 show the PDR for CPM packets at the radio and application levels under different DCC configurations for the 3L and 4L scenarios, respectively. When CAM and CPM have the same priority, they achieve nearly the same PDR (comparing Figure 21-Figure 22 and Figure 23-Figure 24). This is the case because DCC does not differentiate between them. However, a higher PDR degradation is observed when CAM and CPM have different priorities, since CPMs are the only packets dropped by DCC due their lower priority (DP3). The PDR obtained without DCC is significantly higher than the one achieved with the Reactive approach, since the Reactive approach suffers from the synchronization problem previously described that generates packet collisions. With the Adaptive approach, the PDR at the radio level is higher than the PDR without DCC for both the 3L (Figure 23) and the 4L (Figure 24) scenarios. However, the PDR at the application level is lower than the PDR without DCC, especially at short distances, and significantly lower when CAMs and CPMs have different priorities.



(a) Reactive (Q2)
(b) Adaptive (Q2)
(c) Reactive (Q10)
(d) Adaptive (Q10)

Figure 23: PDR (Packet Delivery Ratio) for CPM as a function of the distance between transmitter and receiver with the 3L highway scenario.

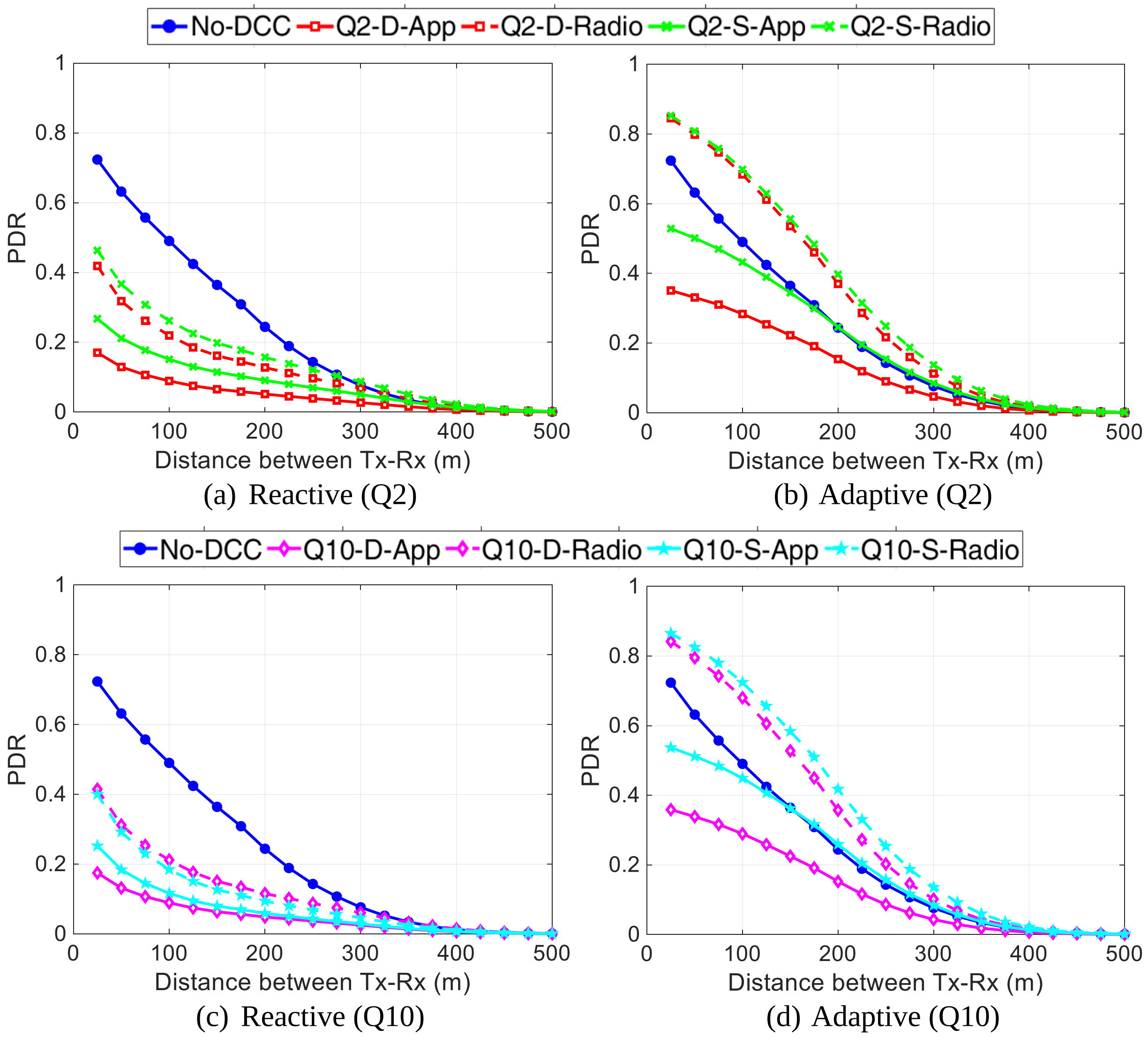


Figure 24: PDR (Packet Delivery Ratio) for CPM as a function of the distance between transmitter and receiver with the 4L highway scenario.

3.2.1.2.3 Perception capabilities

This section analyses the perception capabilities of vehicles for different DCC configurations using only the CPMs. To this aim, different metrics have been defined. One of the most important ones is the Object Perception Ratio, defined as the probability to detect an object (vehicle in this study) through the reception of a CPM with information about it in a given time window. We consider that an object is successfully detected by a vehicle if it receives at least one CPM with information about that object within a given time window. The selection of the appropriate time window for this analysis is important and hence it is estimated considering the following case. In the selected traffic densities, the objects (vehicles in this case) move at an average speed of 50 km/h (13.8 m/s) and need 2.9 s to move 4 m. *T_GenCpm* is defined as a multiple of 100 ms. Therefore, the information about an object is included in a CPM every 300 ms for the 3L and 4L traffic densities. These calculations are important to select the adequate observation time window and correctly evaluate the performance and effectiveness of the collective perception service. As a result, for evaluating the perception in this study, the observation time window of 300 ms is considered. This value corresponds to the time required by ETSI CPM generation rules for a vehicle to send an update about an object in a CPM for the two traffic densities, but could be different for other densities if the vehicles speed significantly change. More information about the selection of the right time window is explained in our TransAID deliverable D5.2 [41].

Figure 25 depicts the average Object Perception Ratio as a function of the distance between the detected object and the vehicle receiving the CPM for different DCC configurations under the two traffic densities. The results show that the No-DCC configuration achieves higher perception (close to 100%) up to 200 m. Even with lower PDR, higher perception is achieved at smaller distances due to higher number of transmitters reporting the same object. Even with higher packet collisions there is still a packet that contains the object information which is successfully received within the selected time window. The degradation of perception with larger distances is observed because of lower PDR levels. It is important to remember that the PDR is calculated as a function of the distance between two vehicles (originating and receiving vehicles), but the Object Perception Ratio accounts for objects detected by any vehicle. In this context, the Object Perception Ratio might refer to an object that is 300 m away from the vehicle receiving the CPM, and the vehicle detecting the object and sending the CPM might be at 250 m, for example. The results obtained in Figure 25a and Figure 25c show that the Reactive approach significantly degrades the perception compared with No-DCC. This is due to the synchronous transmission problem that provokes packet collisions and the high number of packets dropped by DCC. The Reactive approach could therefore negatively impact on the vehicular applications that rely on the CPM information. The results reported in Figure 25b and Figure 25d show that the Adaptive approach achieves very high perception for close critical distances (e.g. up to 200 m) despite having packet drops. Slightly higher perception levels are achieved when both CAM and CPM have the same priority (green and cyan colours), due to the lower number of CPMs dropped. In fact, when both have the same priority, the perception achieved with Adaptive can be higher than the perception achieved without DCC for the 4L scenario considering distances beyond 200 m. When compared with different queue lengths, shorter queue length tends to perform better due to their better PDR.

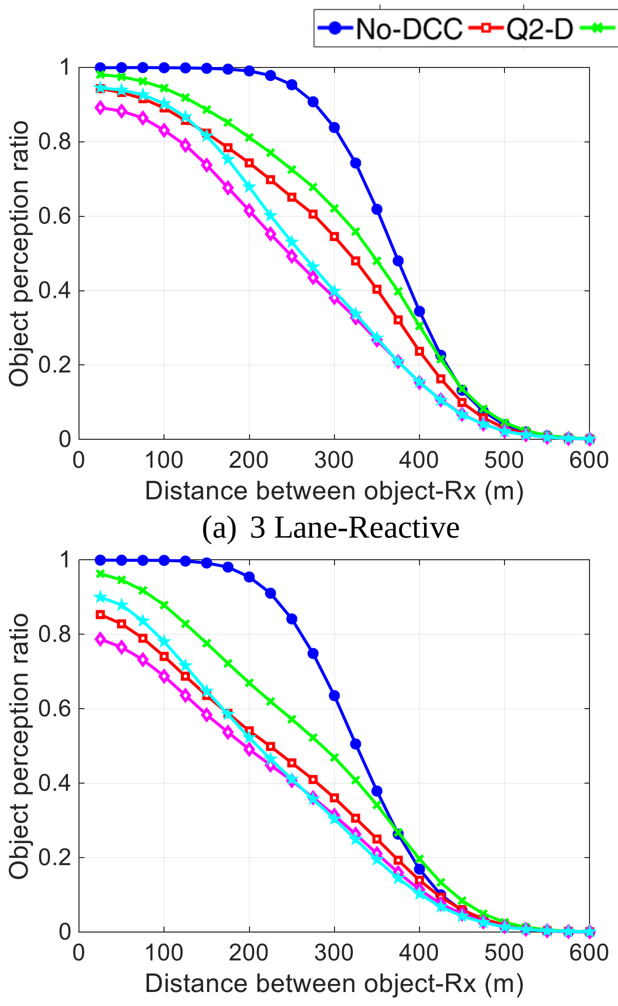


Figure 25: Object Perception Ratio as a function of the distance between the detected object and the vehicle receiving the CPM.

Since the Object Perception Ratio is calculated considering all CPM messages received within the time window, it is also interesting to analyse the frequency of CPMs received that contain the same object information. This information can be considered redundant (and sometimes not necessary), but redundant transmissions help improve the Object Perception Ratio. Figure 26 illustrates the average number of updates received per vehicle under the selected time window (0.3s) about the same object through the reception of CPMs. This metric is referred to as detected object redundancy and is depicted as a function of the distance between the object and the vehicle receiving the CPM for both traffic densities under different DCC configurations. The degradation with distance observed in Figure 26 is a direct consequence of the PDR degradation reported in Figure 23 and Figure 24. Figure 26a and Figure 26c show that the Reactive approach significantly reduces the object redundancy when compared with the No-DCC configuration. It is to be noted that one object update per time window is sufficient to achieve 100% perception. The Reactive approach produces an average of 4 updates per time window for short distances, but still has lower perception due to its lower PDR. Figure 26b and Figure 26d show that the Adaptive approach has a slightly higher object redundancy than No-DCC, especially for higher traffic densities and distances beyond 100 m, when CAM and CPM have the same priority.

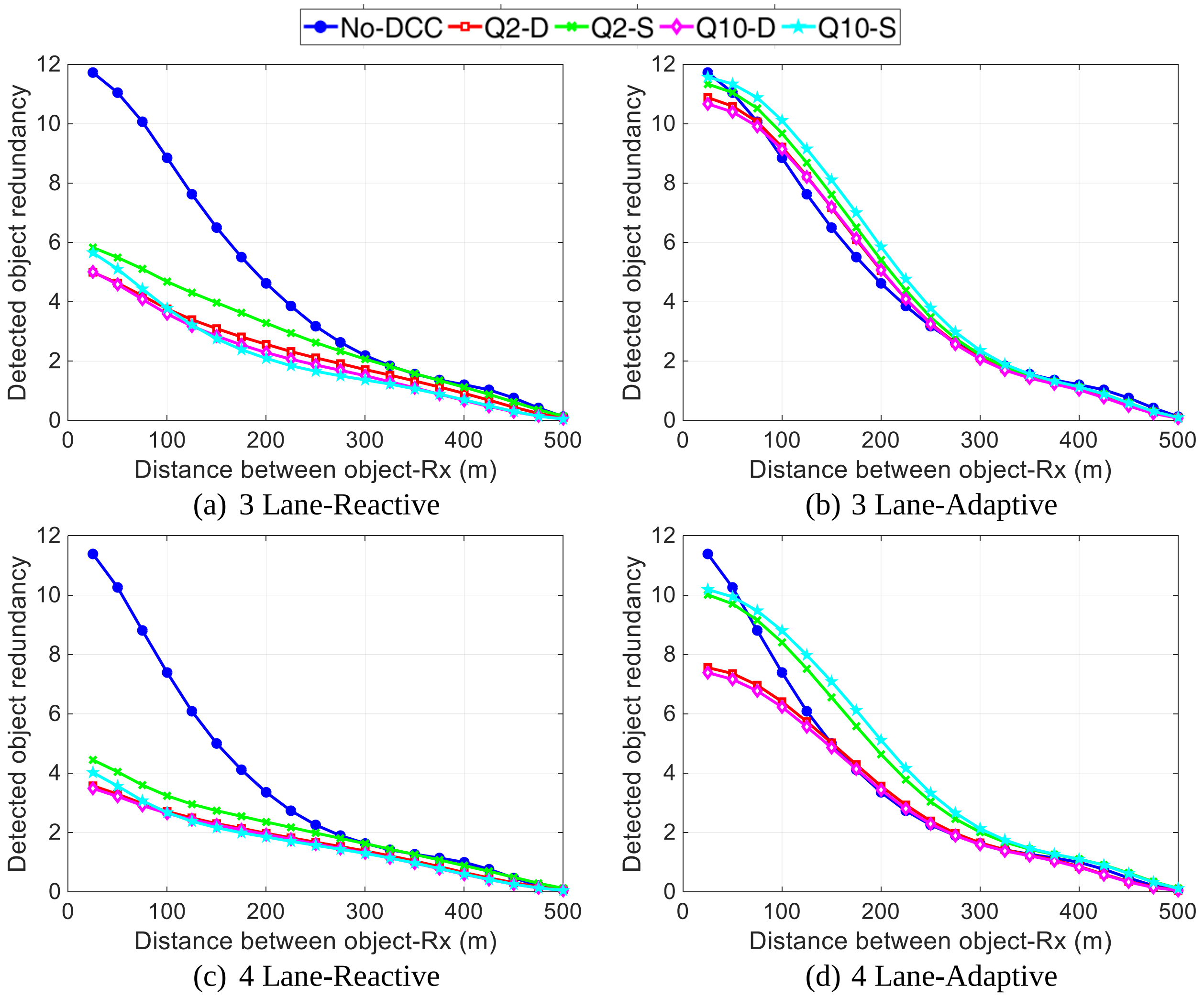


Figure 26: Detected object redundancy as a function of the distance between the detected object and the vehicle receiving the CPM

Connected automated vehicles should regularly receive updates about nearby objects. This can be evaluated through the observation of the time interval between successive reception of CPMs that contain information about each object. Figure 27 and Figure 28 plot this metric, which will be referred to as the time between object updates, for the 4L scenario (similar trends have been observed in the 3L scenario). The metric is represented as a function of the distance between the object and the vehicle receiving the CPMs. The figure shows the average time between updates, and also the 5th and 95th percentiles (vertical lines). Figure 27 shows the time between updates for the No-DCC configuration and the Reactive approach. The results obtained demonstrate that the Reactive approach increases the time between updates when compared with the No-DCC at short distances. This increase is significant when the Reactive approach is configured with higher queue length or when the CPM is configured with lower priority (DP3). This could result in time intervals of around 0.5 seconds for objects at very short distances, which is significantly higher than the time intervals achieved without DCC (in the order of 60 ms). Figure 28 is analogous to Figure 27, but shows the Adaptive approach. The results obtained show that the Adaptive approach increases the time between updates when compared with the No-DCC configuration. However, the Adaptive approach can significantly reduce the time between updates compared to the Reactive approach. Similar to the Reactive approach, the time between updates is higher with Adaptive when it is configured with higher queue length or when the CPM is configured with lower priority (DP3). As both DCC configurations (Reactive and Adaptive) are observed to increase the time between updates, the driving applications that rely on the CPM information could be negatively impacted.

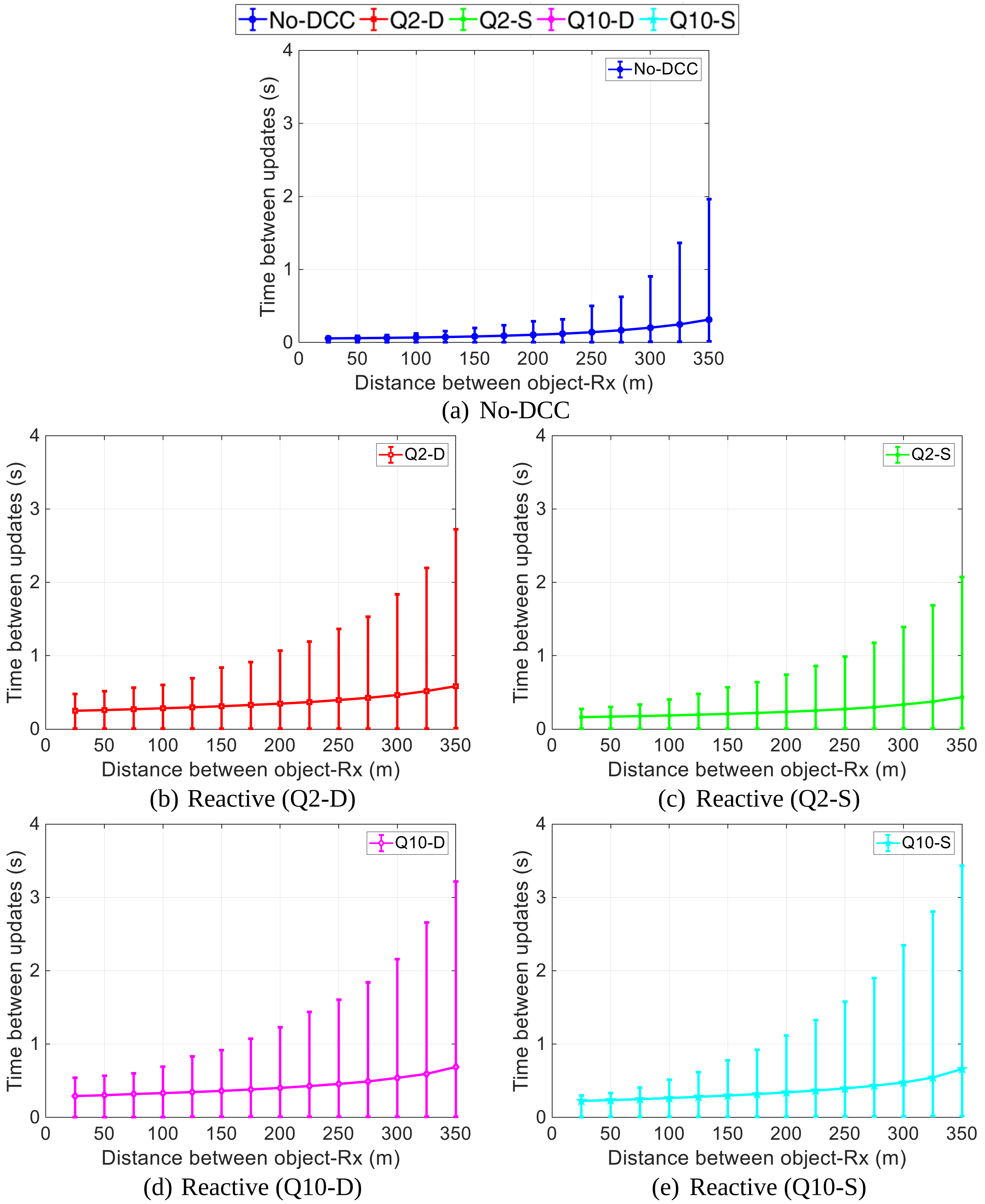


Figure 27: Time between object updates as a function of the distance between the detected object and the vehicle receiving the CPM for 4L traffic density.

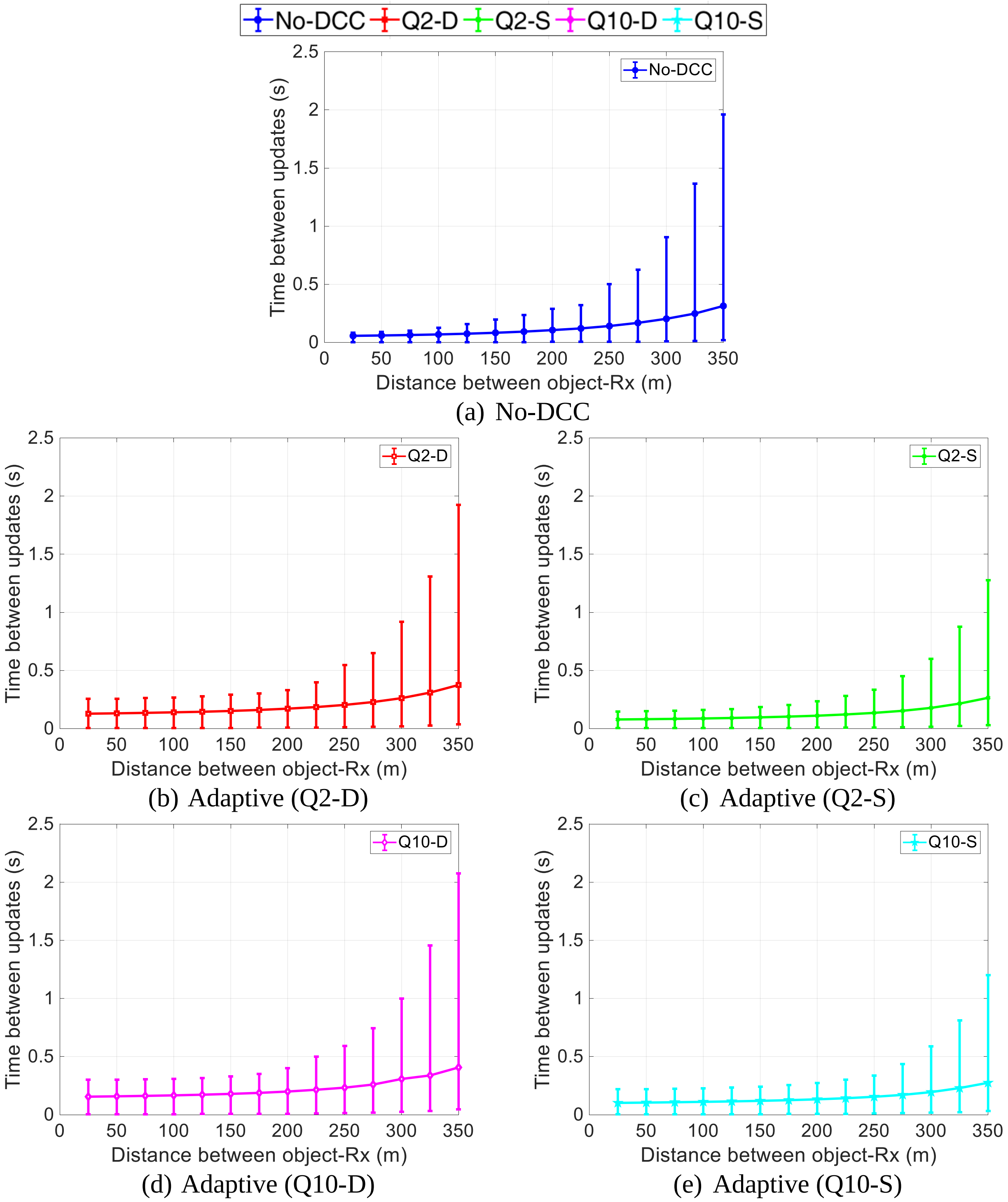


Figure 28: Time between object updates as a function of the distance between the detected object and the vehicle receiving the CPM for 4L traffic density.

To better compare the different configurations, Figure 29 shows the time between object updates for different distance limits between the detected object and vehicle receiving the CPM for the 4L scenario. The distance limits are referred to as critical (0m to 150m), medium (150m to 300m) and high (300m to 450m). Every bar shows the corresponding mean, 5th and 95th percentiles. Figure 29a shows that the No-DCC configuration achieves the lowest time between updates for the critical distances. Figure 29b and Figure 29d show the results for the Reactive approach for different queue lengths. These results show that the Reactive approach increases the time between updates for the critical, medium and high distances for all queue length configurations. Figure 29c and Figure 29e show the time between updates achieved by the Adaptive approach for different queue lengths. The results confirm that the Adaptive approach performs better than the Reactive approach for the different distances. When compared to No-DCC (Figure 29a), the Adaptive approach has higher times between updates due to high packet drops for all the distance limits when both CAM and CPM have different priorities (Figure 29c). When they have the same priority, the Adaptive approach (Figure 29e) has higher times between updates than No-DCC for the critical distances, but lower times between updates for higher distances. Finally, the Adaptive approach with queue length 2 has lower times between updates than the No-DCC configuration.

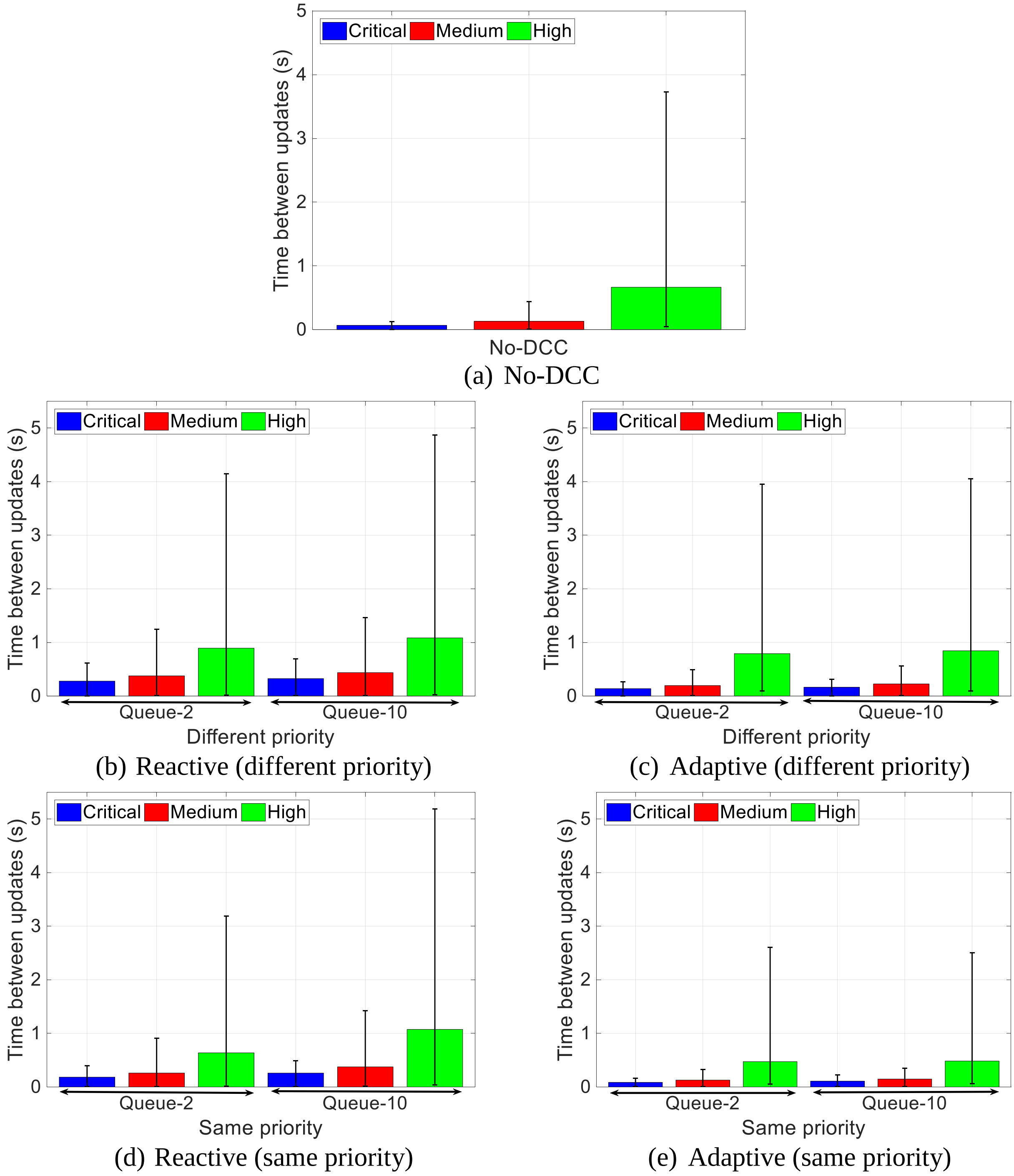


Figure 29: Average time between object updates for different distance limits between the detected object and vehicle receiving the CPM for 4L traffic density.

3.2.1.2.4 Information Utility

The value of cooperative sensing depends on how timely or fresh the information received about the detected objects is. The previous results have shown the time between consecutive updates but did not look into the contents of each CPM. Evaluating this information is important because a vehicle cannot base its driving decision on outdated information. To this aim, Figure 30 shows the information age which is defined as the average time difference between the time the CPM is generated and the time the CPM has been received (we limit the reception distance to 350 m, but it does not have a significant impact on the results). Note that Figure 30a and Figure 30b are shown with Y-axis limits of 0.6 s and Figure 30c and Figure 30d have Y-axis limits of 1.0 s for better readability. The results obtained show that both Reactive and Adaptive approaches significantly increase the information age when compared with the No-DCC configuration. This is because when DCC is not used all the generated CPMs are immediately transmitted and the information age includes only the channel access and transmission delay. With DCC (both Reactive and Adaptive approaches), the generated CPMs must wait in the queue before the gatekeeper opens the gate for transmission. This waiting time in the queue causes the information to be outdated. The delay can range from milliseconds up to the maximum time limit of TTL (1 s in our study). This could negatively impact the vehicular applications that rely on fresh cooperative perception information. When comparing the DCC approaches, we observe that the Reactive approach presents a higher information age due to its fixed T_{off} configuration (Table 7) that can easily go up to 200 ms. When the queue length increases, the waiting time of a packet also increases, thus increasing the information age. The queue length 10 configuration shown in Figure 30c and Figure 30d has a maximum information age of up to nearly 1 s due to the configured TTL of the packets to 1 s.

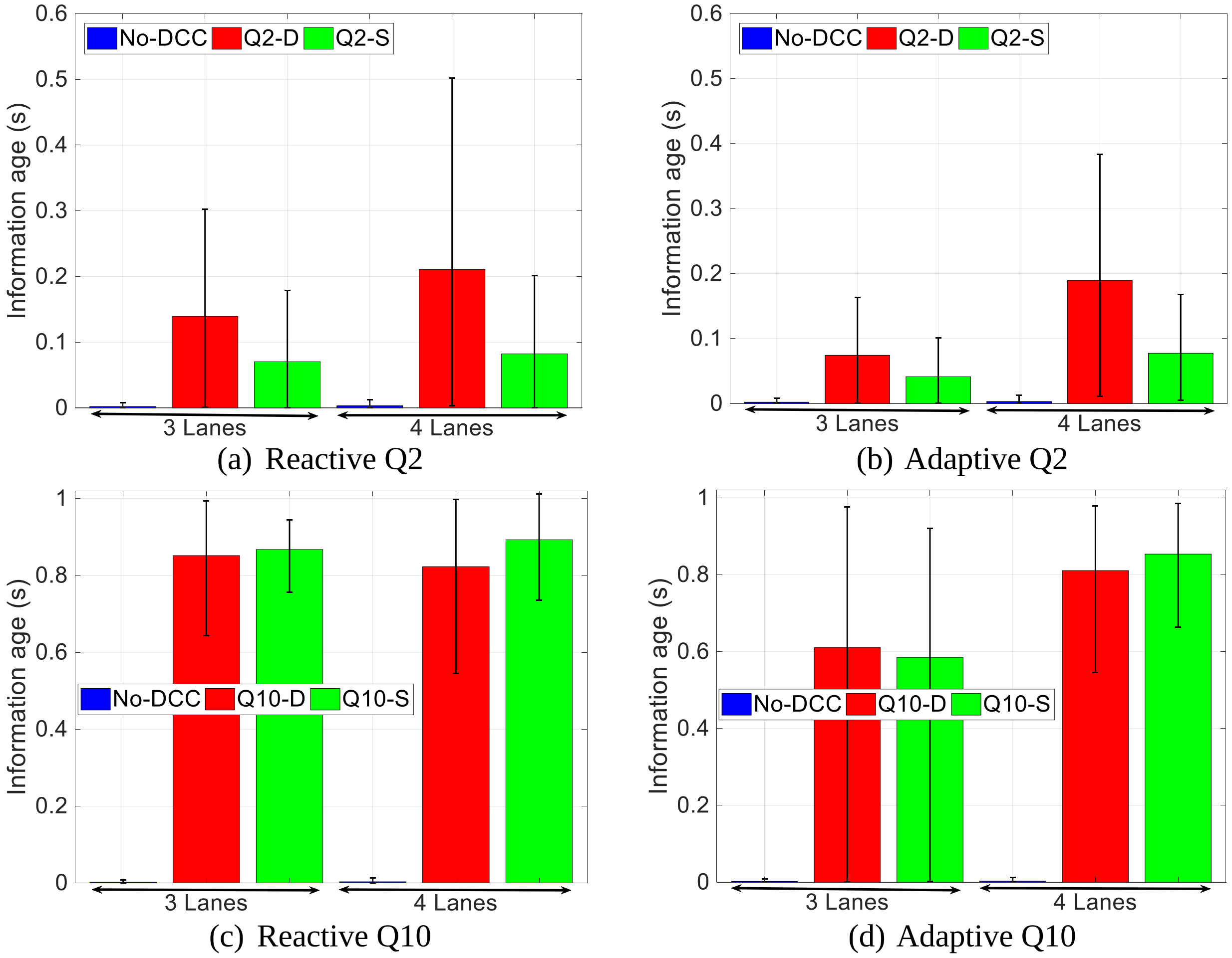
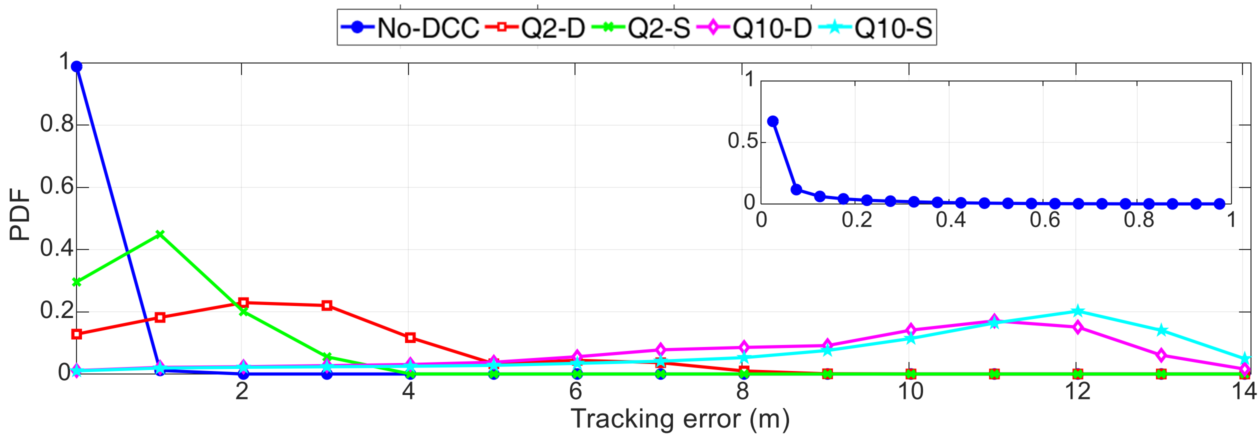
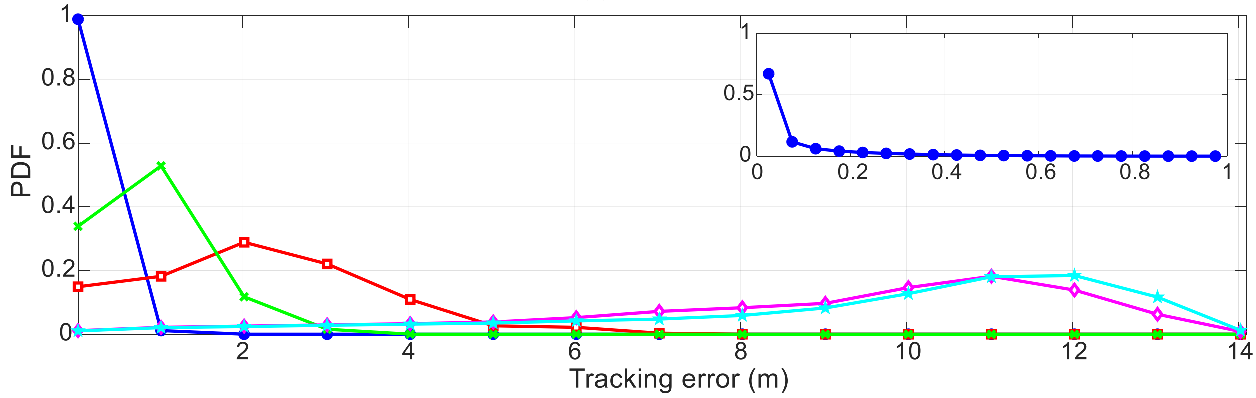


Figure 30: Information age for the 4L scenario.

The information age analysed in the previous figure can have an important impact on perception because the content of the CPM messages is outdated. To show this effect we have computed the tracking error, defined as the difference in the object location between the time the packet is generated and the time the packet has been received. Figure 31 shows the PDF of the tracking error for every CPM received up to 25 m between the detected object and vehicle receiving the CPM for the 4L scenario. The figure shows that the tracking error for both Reactive and Adaptive approaches can be significantly high, especially when considering a queue length of 10 due to the higher information age. However, with the queue length 2 configuration, the tracking error can still go up to 5m. When DCC is not used, the tracking error is below 0.1 m. Sharing outdated information might not add any value to the vehicular applications and thus occupying the channel resources without any actual utility.



(a) Reactive



(b) Adaptive

Figure 31: PDF (Probability Density Function) of the tracking error for every CPM received up to 25m between the detected object and vehicle receiving the CPM for 4L scenario.

3.3 Facilities Layer DCC

The analysis performed in section 3.2 shows that DCC Access can control the channel load and interference generated, thus reducing the number of packets lost due to interferences. However, certain aspects still need to be improved, such as its higher information age due to the waiting queue time. Moreover, DCC Access is not particularly designed to adapt to the different needs of different services generating a different number of messages per second. To try to solve these issues, the Facilities DCC controls the load generated by each application/service, considering the available channel resources and the message rate required by each applications/service.

3.3.1 ETSI solution

The approach described in the current ETSI draft of Facilities DCC [36] controls the message rate of the applications based on the channel resource information feedback from the Access layer DCC and the message generation requirement from applications and services. To this aim, the DCC_FAC obtains the current CBR, the message size and message interval from each application and service, and calculates the minimum interval $T_{off\,min\,ij}$ for each application/service with index j and traffic class with index *i*. Based on this minimum interval, the channel resources are proportionally provided to each application/service and traffic class.

To this aim, each vehicle estimates for each application/service *j* and traffic class *i*, the average message duration $\overline{T_{on\,ij}}$ and the average message interval $\overline{T_{off\,ij}}$ for the latest *n* messages. The average message duration can simply be calculated as the ratio between the average message size of each application/service *j* and traffic class *i* and the data rate (by default, 6Mbps).

Then, the average channel resources consumed by each application can be estimated as:

$$\overline{CRE_{ij}} = \frac{\overline{T_{on\,ij}}}{\overline{T_{on\,ij}} + \overline{T_{off\,ij}}} \quad (7)$$

Using equation (7), the total estimated channel resources CR_i from all applications/services in traffic class *i* can be estimated as:

$$CR_i = \sum_j \overline{CRE_{ij}} \quad (8)$$

Then, the total of available channel resources CBR_a is obtained. This metric is equivalent to the channel resources that the vehicle can use according to the current channel load and specific DCC algorithm (e.g. $CBR_a = \delta$ for the Adaptive approach of DCC Access). This metric is used as the starting point to distribute the available channel resources among the different traffic classes, considering their different priorities. The traffic class with the highest priority is TC_0 and thus the available resources ACR_0 for this traffic class is set equal to CBR_a. If traffic class TC_0 does not consume all the available channel resources, the remaining resources are assigned to the next traffic class. As a result, the available resources ACR_i for traffic class *i* are calculated as:

$$ACR_i = \max(0, ACR_{i-1} - CR_{i-1}) \quad (9)$$

To identify the available channel resources for each application *j* belonging to the same traffic class *i*, the average channel resources consumed by each application is taken into account:

$$ACR_{ij} = \frac{\overline{CRE_{ij}}}{CR_i} \times ACR_i \quad (10)$$

Finally, to calculate the minimum interval $T_{off\,min\,ij}$ for each application/service with index *j* and traffic class with index *i*, the average message size $\overline{T_{on\,ij}}$, and the available channel resources ACR_{ij} are considered:

$$T_{off\,min\,ij} = \overline{T_{on\,ij}} \times \frac{1 - ACR_{ij}}{ACR_{ij}} \quad (11)$$

Finally, the estimated $T_{off\,min\,ij}$ is set as the current $TGen_{ij}$ for application/service with index *j* and traffic class with index *i*.

To illustrate the algorithm with an example, let's consider the scenario where only CAMs and CPMs are transmitted and both belong to the same traffic class (TC2). The Adaptive approach at

DCC Access is used to calculate the total available channel resources, which is assigned to TC2 because there is no other message being transmitted belonging to another traffic class. Therefore:

$$ACR_2 = CBR_a = \delta \tag{12}$$

This equation can be combined with equations (8) and (10) to obtain:

$$ACR_{CAM} = \frac{\overline{CRE_{CAM}}}{\overline{\overline{CRE_{CAM}} + \overline{CRE_{CPM}}}} \times \delta \tag{13}$$

Finally, substituting equation (13) into (11) and using (7), the minimum interval $T_{off\ min\ CAM}$ for the CAM messages can be calculated as:

$$T_{off\ min\ CAM} = \frac{\overline{T_{on\ CAM}}(1-\delta) + (\overline{T_{on\ CAM}} + \overline{T_{off\ CAM}}) \frac{\overline{T_{on\ CPM}}}{\overline{\overline{T_{on\ CPM}} + \overline{T_{off\ CPM}}}}}{\delta} \tag{14}$$

This estimated $T_{off\ min\ CAM}$ is set as the current *TGen* for the CAM messages. An analogous equation can be derived for the CPM.

Let's now expand this example considering that the average size of the latest CAMs is 200 Bytes and the average size of the latest CPMs is 300 Bytes. The time interval between the latest CAMs is 0.2 s and 0.1 s for the latest CPMs. The total available channel resources is $CBR_a = \delta$ = 0.005, which means that each vehicle can use the 0.5% of the channel bandwidth. Assuming a data rate of 6 Mbps, the average channel resources consumed by the CAMs is CRE_{CAM}=0.0013 and by the CPMs is CRE_{CPM}=0.004, obtained with equation (7). These two values can be used to calculate the total channel resources from both CAMs and CPMs (in this example we consider that both belong to the same traffic class), which results in *CR*=0.0053, with equation (8). Then, we can use equation (10) to calculate the available channel resources for the CAM: ACR_{CAM}=0.0013 and for the CPM, ACR_{CPM}=0.0037. Equation (11) can finally be used to compute the minimum interval for the CAM as $T_{off\ min\ CAM}$ = 0.2126 and for the CPM as $T_{off\ min\ CPM}$= 0.1063. The same results are obtained if equation (14) was used instead of equations (7)-(11). As a result of the application of the Facilities DCC algorithm, the time intervals of the CAMs and CPMs need to be increased up to the minimum calculated to reduce the load.

3.3.2 TransAID proposal

The current solution for the Facilities DCC defined at ETSI proportionally shares the channel resources among the available applications/services. While this can be considered a fair approach from the communications perspective, we should take into account that some applications/services could require a certain message rate independently of the specific message size they use. For example, CAMs or CPMs could require that their rate is maintained, while their size can be different, and MCMs can have a much larger size because they contain trajectories (see section 2.2). If we proportionally reduce their rate taking into account their size, services relying on longer messages will be penalized because their transmission rate will be significantly decreased. This issue has motivated in TransAID a Facilities DCC solution that proportionally adapts the rate of each application and service when they belong to the same traffic class. To this aim, the message rate of an application/service with index *j* and traffic class with index *i* is adapted as following:

$$R_{ij} = \frac{\overline{R_{ij}}}{\overline{R_i}} R_{max} \tag{15}$$

where $\overline{R_{ij}}$ is the average message rate of the application/service over the latest *n* messages, $\overline{R_i}$ is the total message rate of the applications/services of traffic class i, and R_{max} is the maximum message rate allowed by DCC Access. This equation assumes that the maximum message rate is assigned to the applications/services of traffic class *i*, but could easily be adapted to scenarios with application/services of different traffic classes as in sub-section 3.3.1. To calculate R_{max}, we use as an input T_{off} provided by DCC Access (Adaptive approach):

$$R_{max} = \frac{1}{T_{off}} \tag{16}$$

Finally, the $TGen_{ij}$ of application/service with index *j* and traffic class with index *i* is calculated as:

$$TGen_{ij} = \frac{1}{R_{ij}} \tag{17}$$

To illustrate the proposed solution, let's consider the same example as in sub-section 3.3.1, where vehicles transmit CAMs and CPMs and are classified with the same traffic class. The time interval between the latest CAMs is 0.2 s and 0.1 s for the latest CPMs. And the current T_{off} in the Access DCC is assumed to be 100 ms. Therefore, the average generation rate of CAMs is $\overline{R_{CAM}}$=5 Hz and the average rate of CPMs is $\overline{R_{CPM}}$=10 Hz. The total generation rate is $\bar{R}$=15 Hz. Using equation (16), we can calculate the maximum rate allowed as R_{max}=10 Hz. Using equation (15) we can calculate the rate of CAMs and CPMs as R_{CAM}=3.3 Hz and R_{CPM}=6.6 Hz. These two values are then used to adapt the *TGen* (*TGenCAM = 0.3* and *TGenCPM = 0.15*) of each service following equation (17).

3.3.3 Evaluation

In this study, we have used the ETSI CPS [40] as a baseline to analyse and improve the cooperative sensing reliability through the use of Facilities layer DCC. For the analysis, we have therefore followed the configurations and scenarios explained in section 3.2. In this study, vehicles are equipped with 360º sensors conform section 3.2. By default, we will consider that vehicles generate CAMs and CPMs based on the ETSI generation rules (see [42] and [40]). To this aim, CAMs are generated depending on the vehicle dynamics (e.g. a vehicle generates CAM every 4 m), while CPMs are generated depending on the number of detected vehicles (i.e. traffic density) and on their dynamics (e.g. an object is included in a CPM every 4m). The traffic scenario is an eight-lane highway with 5 km length and a lane width of 4 meters. We simulate a very high traffic density scenario with 4 lanes in each direction (240 veh/km) and vehicles travel at a maximum speed of 50 km/h. The speeds have been selected based on statistics of a typical US highway obtained from the PeMS database [44]. We analysed the lanes speed of the highway for every hour within a single day and took the average speed for each individual lane. Vehicles created in the simulations have the dimension of 4.8 m x 1.8 m. To avoid boundary effects, statistics are only taken from the vehicles located in the 2 km around the centre of the simulation scenario. All vehicles are assumed to be equipped with an ITS-G5 transceiver (100% penetration) and operate in the same channel. The propagation effects are modelled using the Winner+ B1 propagation model following 3GPP guidelines [43]. Regarding the traffic class, both CAMs and CPMs are configured with the same priority DCC profile DP2. With respect to DCC configuration, different approaches are analysed:

1. No-DCC. This solution does not consider any DCC protocol. Therefore, it only considers the ETSI CPS with its basic CPM generation rules.
2. DCC Access. This solution implements the Adaptive approach of DCC Access, since it is the approach that provides better performance, according to the results obtained in previous sections.
3. DCC Facilities (δ). This solution considers the Adaptive approach of DCC Access and the ETSI Facilities DCC described in sub-section 3.3.1.
4. DCC Facilities (T_{off}). This solution implements the Adaptive approach of DCC Access and the TransAID proposal for the Facilities DCC described in sub-section 3.3.2.
5. RMLA. This solution does not implement any DCC protocol. Instead, it implements the Redundancy Mitigation & Look-Ahead mechanisms proposed in TransAID and explained in detail in the deliverable 5.2 [41].
6. RMLA+DCC Facilities (T_{off}). This solution adds the TransAID proposal for the Facilities DCC described in sub-section 3.3.2 to RMLA.

The goal is to compare their performance and understand whether an intelligent adaptation of the information generated and transmitted with RMLA (without DCC) could outperform the different solutions evaluated with DCC.

Table 15 shows the average generated and transmitted CAMs and CPMs for different approaches under the 4-lane scenario and 360º sensor configuration. The table shows that without DCC, the No-DCC and RMLA approaches have the same generation and transmission rates because all messages generated are transmitted. When comparing between No-DCC and RMLA approaches, the RMLA significantly reduces the generation rate by 66% thanks to the efficient generation of CPMs and the inclusion of only the necessary objects in each CPM. For all DCC enabled configurations, the transmission rate is lower than the generation rate for both CAM and CPM because of the packet drops performed by the Access layer DCC. With only DCC Access, the transmission rate is significantly reduced increasing the packet drops as shown in the table. When Facilities layer DCC is incorporated, it reduces the generation rate itself by controlling the *TGenCAM* and *TGenCPM* thus reducing the packet drops significantly.

Table 15: Average generated (Gen.) and transmitted (Tx.) CAMs and CPMs

Configuration	CAM			CPM		
	Gen.	**Tx.**	**Diff.**	**Gen.**	**Tx.**	**Diff.**
No-DCC	3.3 Hz	3.3 Hz	0%	9.6 Hz	9.6 Hz	0%
DCC Access	3.3 Hz	2.1 Hz	-36.4%	9.6 Hz	6.0 Hz	-37.5%
DCC Facilities (δ)	2.3 Hz	2.0 Hz	-14.4%	5.1 Hz	4.1 Hz	-18.1%
DCC Facilities (T_{off})	1.5 Hz	1.2 Hz	-24.0%	4.9 Hz	4.6 Hz	-6.88%
RMLA	3.3 Hz	3.3 Hz	0	3.2 Hz	3.2 Hz	0
RMLA+DCC Facilities (T_{off})	2.7 Hz	2.6 Hz	-0.4%	2.8 Hz	2.7 Hz	-2.9%

Figure 32 represents the PDF of the number of objects included in each CPM for the different solutions evaluated. The figure shows that the No-DCC implementation and DCC Access include the same number of objects in each CPM because packet dropping without any feedback does not alter the CPM generation rules (see Table 15). It is also observed that these two solutions generated higher number of small CPMs with a low number of objects. Frequent transmission of small CPM messages adds significant signalling overhead. The figure also shows that RMLA generates longer CPMs while minimizing the number of CPMs that contain a small number of objects. While this increases the message size, it reduces the channel overhead and improves the efficiency. Both RMLA and RMLA+ DCC Facilities (T_{off}) include objects in a CPM in a similar way, but a lower number of smaller CPMs are generated in RMLA. The comparison between DCC Facilities (δ) and (T_{off}), shows the number of small CPMs is reduced with DCC Facilities (δ). The obtained results clearly show the potential benefits of RMLA to reduce the signalling overhead.

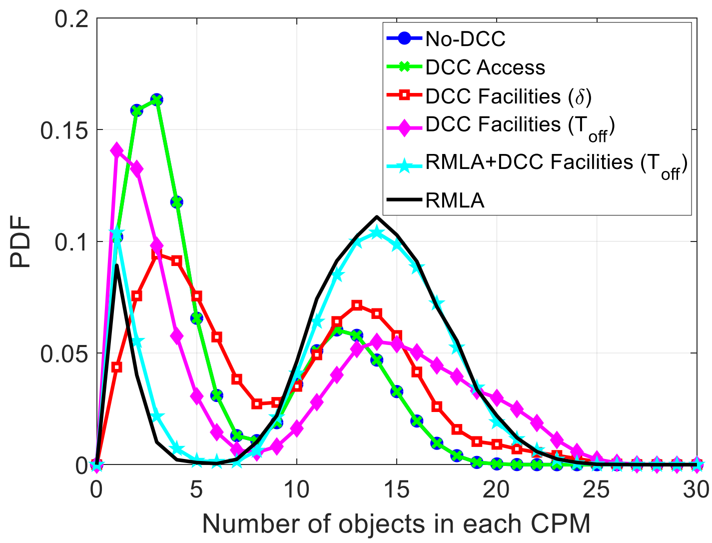


Figure 32: PDF (Probability Density Function) of the number of objects included in each CPM for different approaches under the 4-lane scenario and 360º sensor configuration.

The impact of the message transmission rate and the number of objects included in each CPM has an impact on the channel load. Table 16 shows the average Channel Busy Ratio (CBR) for the different solutions evaluated. Without DCC, the average CBR experienced was 75%. As shown in the table, RMLA and DCC based approaches control the CBR and hence the interferences generated. However, the lowest CBR is achieved with the RMLA solution. The main difference between the RMLA and DCC based approaches is that, RMLA reduces the channel load by controlling the inclusion of objects in each CPM and its generation rate at the CPS. With RMLA, all generated CPMs are transmitted. However, in the DCC based approaches, messages can be dropped at the Access layer and the information they contain can therefore be lost. Note that when DCC is incorporated with the RMLA approach, the CBR increases when compared with RMLA. This is because some CPMs generated by RMLA are dropped by DCC at the Access layer, which increases the generation of small CPMs (see Figure 32). This increases the channel overhead, thereby increasing the CBR. When comparing DCC Facilities (δ) and (T_{off}), a CBR reduction is observed with DCC Facilities (T_{off}).

Table 16: Average CBR (Channel Busy Ratio)

Configuration	CBR	Difference
No DCC	75.0%	0%
DCC Access	61.9%	-17.5%
DCC Facilities (δ)	61.62%	-17.8%
DCC Facilities (T_{off})	61.00%	-18.7%
RMLA	55.52%	-26.0%
RMLA+DCC Facilities (T_{off})	57.46%	-23.4%

To evaluate the reliability achieved by the different solutions evaluated, we have measured the Object Perception Ratio. It is defined as the probability to detect an object (vehicle in this study) through the reception of a CPM with information about it in a given time window. Following the analysis performed in section 3.2.1.2.3 and in our work D5.2 [41], an observation time window of 300ms is considered. This value corresponds to the time required by ETSI CPM generation rules for a vehicle to send an update about an object in a CPM for the selected traffic densities and is just used for this analysis (it does not affect the generation rules). Figure 33 depicts the average Object Perception Ratio as a function of the distance between the detected object and the vehicle receiving the CPM. The results obtained show that all approaches achieve high perception, close to 100%, up to 200m. The best performance is achieved by RMLA because it selects only the necessary objects that need to be shared in the network which reduces the CPM generation rate (Table 15) and the CBR (Table 16). This higher perception is achieved in RMLA without using DCC. When DCC is incorporated with RMLA, the perception tends to degrade because the DCC Facilities reduces the generation rate. The comparison between DCC Facilities (δ) and (T_{off}) reveals that the solution proposed in TransAID (T_{off}) improves the performance of the ETSI DCC Facilities draft because of its lower CBR (Table 16) and higher CPM transmission rate.

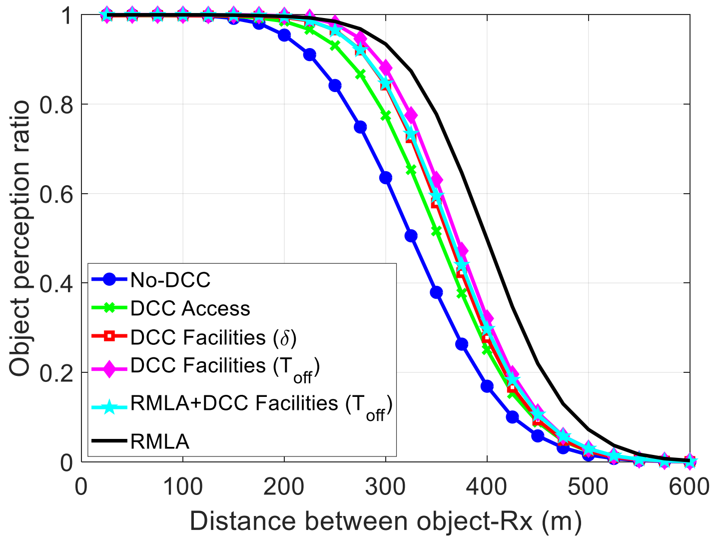
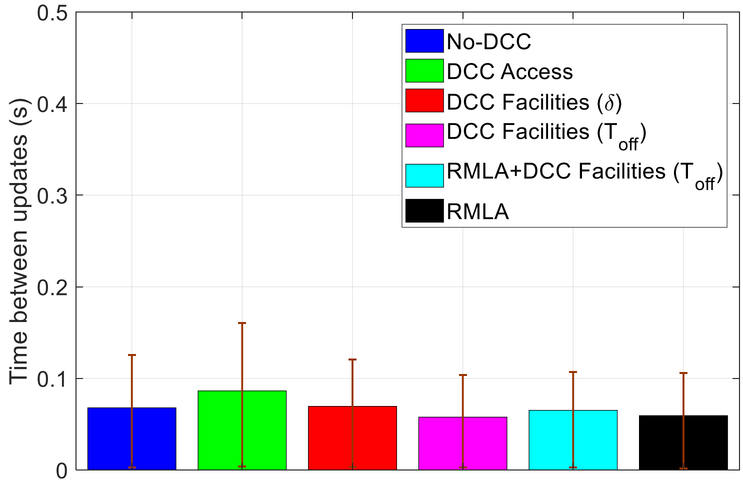


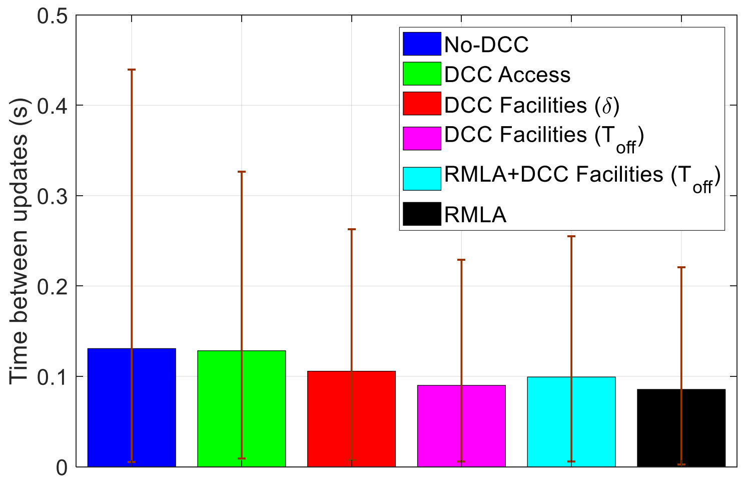
Figure 33: Object Perception Ratio as a function of the distance between the detected object and the vehicle receiving the CPM.

Connected automated vehicles should regularly receive updates about nearby objects. This can be evaluated through the observation of the time interval between successive reception of CPMs that contain information about each object. Figure 34 shows the time between object updates for different distance limits between the detected object and vehicle receiving the CPM. Figure 34a shows the time between updates for the critical distance limits (0m to 150m); Figure 34b shows the time between updates for the medium distance limits (150m to 300m) and Figure 34c shows the time between updates for the high distance limits (300m to 450m). Every bar shows the corresponding mean, 5th and 95th percentiles. Figure 34a shows that all solutions provide object updates below 200ms for the critical distances. For medium and high distances, the time between updates are significantly increased for the No-DCC approach, as shown in Figure 34b and Figure 34c. The figures also show that RMLA provides more frequent updates than other approaches reducing the time between updates significantly. This once again justifies the higher perception achieved by RMLA for medium and higher distances.

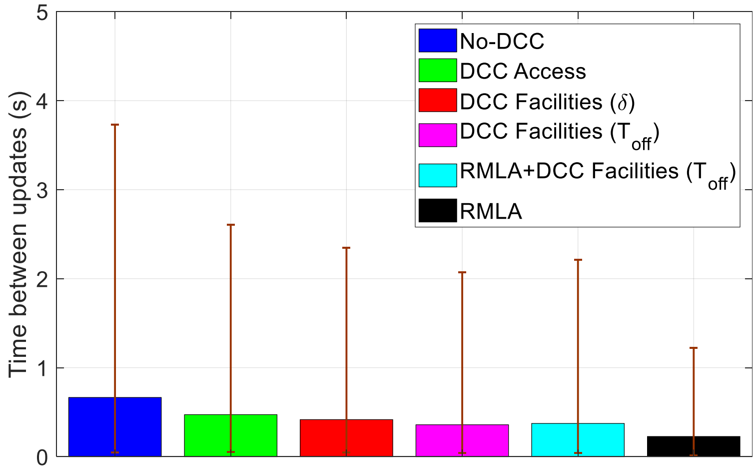
The value of cooperative sensing depends on how timely or fresh the information received about the detected objects is. The previous results have shown the time between consecutive updates but did not look into the contents of each CPM. Evaluating this information is important because a vehicle cannot base its driving decision on outdated information. To this aim, Figure 35 shows the information age which is defined as the average time difference between the time the CPM is generated and the time the CPM has been received (we limit the reception distance to 350m, but it does not have a significant impact on the results). Every bar shows the corresponding mean, 5th and 95th percentiles. The results obtained show that the DCC based approaches significantly increase the information age when compared with the No-DCC solution and RMLA. This is because when DCC is not implemented, all the generated CPMs are immediately transmitted and the information age includes only the channel access and transmission delay. With DCC, the generated CPMs must wait in the queue before the gatekeeper opens the gate for transmission. Even though the facilities DCC alters the message generation rate, the waiting time in the queue persist. However, both DCC Facilities (δ) and (T_{off}) reduce the information age when compared with the DCC Access approach. It is also interesting to analyse the RMLA+DCC Facilities (T_{off}) solution in detail. Despite the queuing delay, this solution significantly reduces the information age when compared with other DCC approaches. Thanks to the reduction of the CPM generation rate by both RMLA and the T_{off} approaches. Since the CPM generation rate is greatly reduced, it reduces the CBR and most of the time the packets are transmitted immediately without waiting in the queue. However, from the higher difference in 95th percentile it is clear that the packets waiting time is highly oscillating with this approach.



(a) Critical distances.

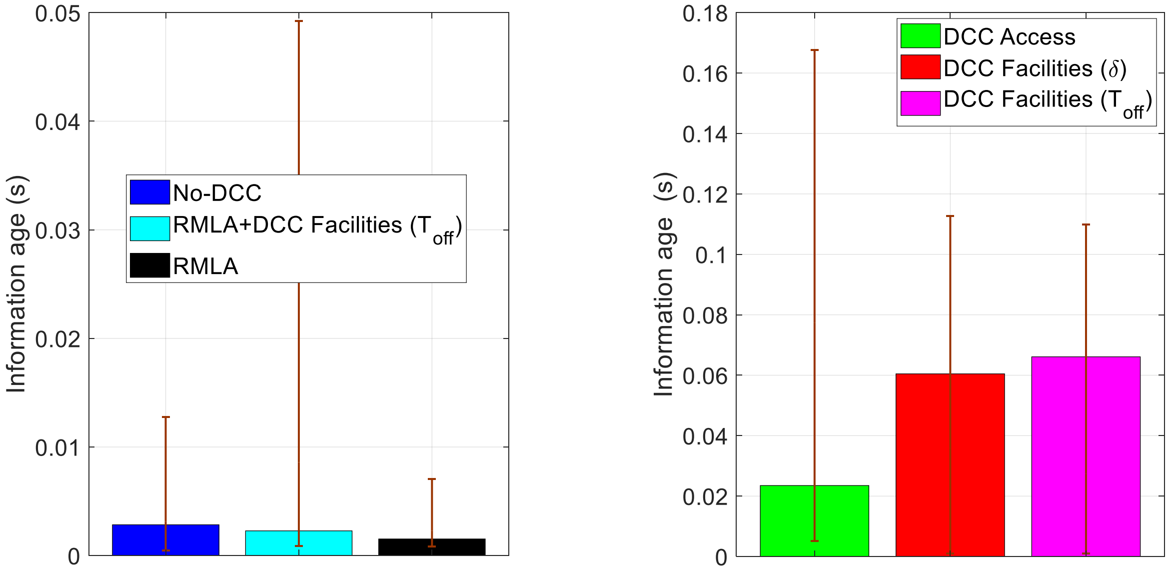


(b) Medium distances.



(c) High distances.

Figure 34: Average time between object updates between the detected object and vehicle receiving the CPM for different approaches under the 4-lane scenario and 360º sensor configuration.



(a) **(b)**

Figure 35: Information age for different approaches under the 4-lane scenario and 360º sensor configuration.

4 Context-based Acknowledgement for Cooperative Broadcast V2X messages

Connected vehicles rely on V2X (Vehicle to Everything) communication technologies to expand their capabilities by utilizing contextual information beyond the local environment they can perceive. Based on the Car 2 Car Communication Consortium (C2C-CC) roadmap published in [50], the first phase in the deployment of V2X communications (a.k.a. Day1) will include Cooperative Awareness (CA) and Decentralized Environment Notification (DEN) services to disseminate the vehicles' status information (location, speed, acceleration, heading direction, etc.) as well as the occurrence of dangerous situations (road work, hard-breaking, etc.). These services are supported by the exchange of broadcast CA and DEN messages (i.e. CAM and DENM) and are tailored for active safety and traffic efficiency applications.

Connected automated vehicles (CAVs) can further improve their perception of the surrounding environment by exchanging sensor information (either raw data or processed data in the form of detected objects) with neighbouring vehicles. The sharing of detected objects among CAVs will be performed by the Collective Perception Service (CPS) and will enable part of the Day2 applications and services [50]. The CPS is also under standardization at ETSI where its Technical Committee on ITS (Intelligent Transportation Systems) is defining, for instance, the format of the messages to be used for the exchange of the sensor information (known as the Collective Perception Message - CPM), and their generation rules. Note that TransAID is taking an active role in the standardization of the CPS and has participated in the evaluation and analysis of the CPM generation rules. Additional details of this study can be found at the TransAID deliverable on V2X-based cooperative sensing and driving in Transition Areas [47] and the ETSI Technical report on the Analysis of the Collective Perception Service (CPS) [26].

Day3+ applications and services [50] will look at a step beyond by allowing CAVs to coordinate their manoeuvres. This Manoeuvre Coordination Service (MCS), that is being thoroughly analysed in TransAID, also incorporating the support of the infrastructure to this service, will support the exchange of planned trajectories/routes and manoeuvre intentions among CAVs, and the generation of advices from the infrastructure to assist in this coordinated manoeuvring process. The exchanged information between the vehicles, or between the vehicles and the infrastructure, is performed using broadcast MCM messages.

As it has been highlighted above, most of the V2X applications and services will be supported by the broadcast exchange of V2X messages between vehicles (Vehicle-to-Vehicle, V2V), and between vehicles and the infrastructure (Vehicle-to-Infrastructure, V2I). In all these V2X apps and services, all nearby vehicles and infrastructure nodes might be interested in the transmitted V2X messages and therefore most of them are transmitted in broadcast mode. In addition, broadcast transmissions do not require identifying the neighbouring vehicles to contact them, which enables a fast exchange of information. This is particularly of interest in vehicular networks due to their dynamics and frequent topology changes. While unicast messages have their own mechanisms to assure their correct delivery, which is based on feedbacks/acknowledgments from the receiver, to date there are no such mechanisms for broadcast messages. Ensuring a high reliability in the delivery of V2X broadcast message is a challenge and essential for critical V2X applications/services.

For example, for the MCS, the exchange of planned trajectories and the vehicles' dynamics would avoid wrong predictions about the driving intentions of the vehicles and can therefore contribute to improve the traffic safety and efficiency. To make it more precise to a particular situation, CAVs on a roundabout could share their planned manoeuvres so that CAVs waiting to enter the roundabout know in advance whether they can enter or not. However, vehicles should not take any actions that might depend on the correct delivery of the MCM messages and the acknowledgement of the cooperative manoeuvring process.

Extending the examples to the CPS, consecutive failures in the reception of CPM messages from neighbouring vehicles would reduce the awareness of the driving environment. This might provoke dangerous driving situations. In the event of a crossing pedestrian, certain vehicles approaching the pedestrian might not detect the person using their built-in sensors. It would be of great value if these vehicles could be informed about the presence of this pedestrian. This could be performed by other vehicles that can detect the pedestrian and ensuring the delivery of the CPM messages.

The motivation to design mechanisms that contribute to increasing the reliability of cooperative V2X broadcast messages is not only dependent on the added value for the V2X applications and services. Recently, there is a growing interest in this field of research due to the new 5.9 GHz band-plan in US [51]. This band-plan might result in that the 5.9 GHz band is not fully reserved for ITS services, and V2X and WiFi technologies would have to share it. In this case, the reliability of V2X broadcast transmissions might be compromised due to the increased channel usage. In this context, V2X applications and services designed to support active safety and critical/risky driving situations might need to rely on mechanisms that can ensure the correct delivery of the exchanged information using the cooperative V2X messages.

To address these issues, current V2X communication technologies operating at 5.9 GHz are incorporating mechanisms to enhance the reliability of broadcast transmissions. For example, V2X communications based on cellular technologies (i.e. LTE-V and 5G NR) perform multiple transmissions of the same packet and exploit the HARQ (Hybrid Automatic Repeat Request) mechanism to combine the retransmissions and increase the likelihood to correctly receive the packet [52]. IEEE 802.11bd, the amendment to the IEEE Std 802.11 standard operating at 5.9 GHz (a.k.a. IEEE 802.11p), is also being designed to support higher reliability for V2X broadcast transmissions. To this aim, IEEE 802.11bd is leveraging the MAC/PHY mechanisms that have been developed for the IEEE 802.11 technology during the past decade and are part of the IEEE 802.11n, 802.11ac, 802.11ax amendments. At the PHY layer, this includes for example the support for midambles to achieve a better estimation of the channel and combat the Doppler effect, and advanced coding schemes such as LDPC (Low Density Parity Check). At the MAC layer, IEEE 802.11bd will support adaptive repetitions of V2X broadcast messages, with the number of repetitions varying based on the measured load of the V2X channel [53]. IEEE 802.11bd has also analysed the option to create A-MPDU/A-MSDU (Aggregated MAC Packet/Service Data Unit) in order to improve the efficiency of the radio channel. In the context of vehicular networks, A-MPDU/A-MSDU would allow, for example, to transmit together multiple V2X packets, or to transmit large V2X messages without fragmenting them. This might be the case, for example, for CPMs messages that could include a large number of objects either with raw or very detailed information.

The analysis conducted so far in TransAID has shown that with the current ETSI specification of the CPS in terms of message format and object description, the size of a CPM is well below the message threshold for fragmentation (i.e. 1600 bytes) [54]. In addition, IEEE 802.11bd has adopted that A-MPDU/A-MSDU will be only available for unicast V2X communications –it did not pass the motion for its use in broadcast V2X communications. Another ongoing discussion under IEEE 802.11bd is whether to consider the use of the multi-frame transmission mechanism for V2X communications. Multi-frame transmission would allow a vehicle to transmit multiple V2X packets simultaneously, with a Short Inter Frame Space (SIFS) between them. Like A-MPDU/A-MSDU, multi-frame transmission would help improve the channel efficiency due to a better utilization of the radio channel. On the other hand, multi-frame transmission would allow to process each packet separately which would allow to detect what packet, out of the multiple ones transmitted, is received correctly (A-MPDU/A-MSDU would treat all packets as a single one). An important concern that is being discussed in the standardization of IEEE 802.11bd is the fairness of V2X applications using A-MPDU/A-MSDU and multi-frame transmission. Vehicles supporting these mechanisms would be granted to use the radio channel to perform transmissions of multiple V2X messages at once, which would not be “fair” to vehicles that do not support them and need to gain access to the channel (e.g. through back-off mechanism) for the transmission of each V2X message.

In addition, in the ongoing discussions for the standardization of IEEE 802.11bd, it has been recently proposed that the use of acknowledgments as a feedback mechanism should be adopted to ensure the correct delivery of V2X broadcast messages. The proposed broadcast acknowledgment mechanism seeks to control the load and signalling that might be caused by the feedback from all neighbouring vehicles that receive a V2X broadcast message requiring acknowledgement. To this aim, the proposed mechanism specifies that the V2X application should indicate the conditions/situations that trigger a request for feedback about the reception status of the V2X broadcast message to the receiving vehicles. The proposed mechanism also suggests that for controlling such load and signalling, the V2X application should also identify what receivers should reply to the broadcast acknowledgement request from the transmitting vehicle. These two decisions are out of the scope of the IEEE 802.11bd specification, and they should be defined based on the V2X application requirements.

Inspired by the discussions presented in IEEE 802.11bd, TransAID proposes and evaluates a context-based broadcast acknowledgement mechanism that is used by the transmitting vehicles to selectively request the acknowledgment of specific (critical) broadcast messages. The proposed mechanism also implements retransmissions if the broadcast messages are not received correctly by the addressed receivers. The proposal has been evaluated for a CPS service that is designed to support the awareness of crossing pedestrians at intersections. The obtained results show that the proposed mechanism can contribute to increasing the reliability in the exchange of CPM messages between vehicles approaching to the intersection, and thus to reduce the potential risk that the vehicles approaching the intersection do not detect the pedestrians. In addition, the proposal is message and service agnostic, and could be used to increase the reliability of any other V2X broadcast message generated by any other applications or services.

4.1 Context-based Broadcast Acknowledgment

One of the major challenges for V2X technologies is to increase the reliability of V2X applications/services that are based on the (periodic) exchange of broadcast messages. Under the ongoing standardization framework of IEEE 802.11bd (a.k.a. Next Generation Vehicular networks, NGV), which is expected to finish by the end of 2021, the use of the context-based broadcast acknowledgment mechanism has been proposed to assure the delivery of V2X broadcast messages. The design of this mechanism is being led by the company Autotalks which has presented in [55] and [56] different proposals to the IEEE P802.11-Task Group BD (NGV) [57]. Section 4.1.1 describes the mechanism that was presented by Autotalks in [58] that finally has been included in the motion booklet for IEEE 802.11bd and has inspired the proposal made by TransAID.

4.1.1 Concept

The context-based broadcast acknowledgement mechanism presented in [58] aims at establishing a feedback loop between the transmitting vehicle and the receiver(s). The feedback loop is used as a report to inform the transmitter about the delivery status of the transmitted V2X broadcast messages. In IEEE 802.11 standards, this feedback loop is performed through the transmission of an acknowledgement (ACK) packet from the receiver to the transmitter. This acknowledgment mechanism has been used in the framework of IEEE 802.11 for unicast data transmissions. The acknowledgment mechanism for unicast has been used as the basis for broadcast data transmissions under the IEEE P802.11-Task Group BD (NGV).

Two important design aspects need to be taken into account in the context-based broadcast acknowledgement mechanism. The first one is related to the fact that multiple vehicles receiving a V2X broadcast message could send back ACK packets to the transmitting vehicle. These ACK packets would interfere with each other if they are not properly coordinated. If they interfere with each other, the transmitting vehicle would not properly receive any of them, which would trigger a retransmission of the broadcast message. To avoid this negative effect, ACK packets could be coordinated, but this is a complex task given the broadcast nature of the messages and can significantly decrease the transmission efficiency.

The second important design aspect is related to the network load and scalability. In general, the channel load and interference increase as the number of vehicles and the data traffic they generate augment. The increase of the interference augments the probability of packet loss and therefore the need to retransmit V2X messages. As a consequence, the interference is further increased due to the increase of message retransmissions. In Systems Theory, this is known as a positive feedback, defined as a process that amplifies changes and tends to move a system away from its equilibrium state and make it more unstable.

To avoid these situations, the context-based broadcast acknowledgment mechanism seeks controlling both: 1) what V2X broadcast messages need feedback; and 2) the receiving vehicle(s) to whom a feedback should be requested. The mechanism proposed to the IEEE P802.11-Task Group BD does not specify any particular conditions or characteristics to identify any of the two. This is typically the case since IEEE P802.11 standards working groups focus on the specification, operation and protocols of lower layers of the OSI reference model (i.e. MAC/PHY), while these two decisions are to be made by the higher layers (i.e. facilities/application) [58].

4.1.2 Proposal

Inspired by [58], this study shows in Figure 36 a graph diagram to integrate the context-based broadcast acknowledgement mechanism in the ETSI ITS protocol stack [59]. The ETSI ITS protocol stack focuses on the higher layers of the OSI reference model, that in ETSI ITS are referred to as Applications, Facilities, and Networking & Transport layers. The ETSI ITS's Access layer integrates the IEEE 802.11 MAC/PHY layers.

At the higher layers (i.e. Application and Facilities layers), the 'V2X App' generates V2X broadcast messages. Upon the generation of each of these messages, the 'V2X App' also checks whether the conditions are met to request an ACK packet from the receiver for this message (see block 'Conditions to request ACK' in Figure 36). As indicated above, these conditions are specific and defined by the 'V2X App', e.g. based on reliability requirements. In case the conditions to request an ACK are met, the 'V2X App' creates a tag that is used to transfer information between layers within the V2X protocol stack and that is attached to the V2X broadcast message. This tag includes the unique ID of the V2X broadcast message that needs to be acknowledged (V2XpktId), and the ID of the receiver that will be requested to reply with the ACK packet (RxId). Then, the V2X broadcast message, together with the created tag, if the conditions are met, is passed down to the lower layers.

When the V2X broadcast packet reaches the MAC layer, it is checked whether it has attached the tag (see 'TAG included?' in Figure 36). In any case, the V2X broadcast packet is passed down to the lower layers for its transmission. In case the V2X broadcast message has attached the tag, a copy of the V2X packet is also saved at the MAC layer (see 'Copy V2X pkt' in Figure 36). Next, the MAC layer creates a Broadcast ACK Request (BAR) packet. This BAR packet is addressed to the receiver RxId identified by the V2X App. It should be noted the BAR packet is unicast and that the transmitter waits a limited time for the reception of the ACK from the addressed receiver (see 'Set timer for receiving ACK' in Figure 36). Once the BAR packet is populated, it is passed down to the lower layers. If the transmitting vehicle receives the ACK packet, it checks whether the ACK packet shows that the addressed RxId vehicle received the V2X broadcast packet with ID V2XpktId (ACK) or not (NACK). It might also happen that the timer set by the transmitting vehicle to wait for the ACK packet expires before receiving any ACK packet. In case the ACK packet indicates NACK or the timer to wait for the ACK packet had expired, the transmitting vehicle retransmits the saved copy of the V2X broadcast packet. The number of retransmissions is limited to *CounterReTx*. The retransmitted V2X broadcast packet also triggers the transmission of the BAR packet, following similar procedures as the original transmission of the V2X broadcast packet. In case there are no remaining retransmissions, or the ACK packet indicates ACK, the MAC layer could send a report to the 'V2X App' that shows the result of the context-based broadcast acknowledgment mechanism.

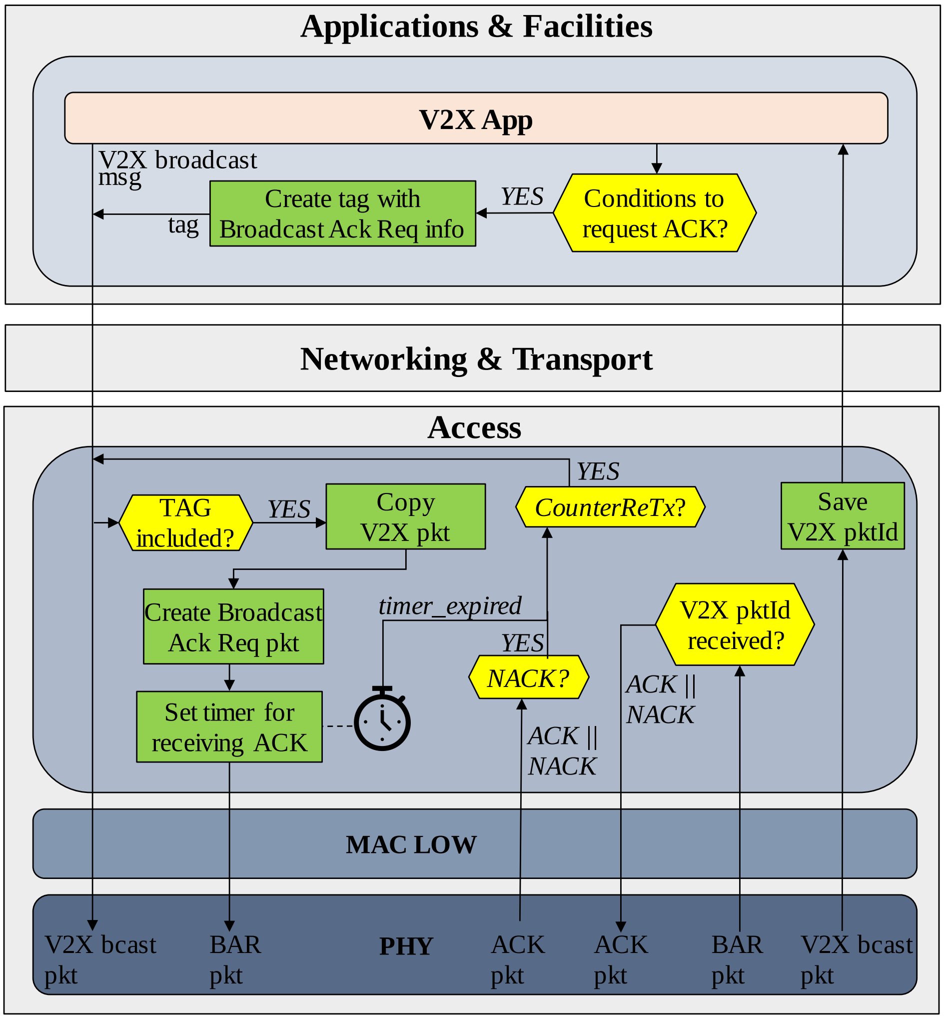


Figure 36: Integration of the broadcast acknowledgment mechanism in the ETSI ITS protocol stack

Figure 36 also shows the operation at the receiving vehicles. Upon the reception of the V2X broadcast packet, the MAC layer saves the ID of the received packet (V2XpktId). This saved V2XpktId is used, in the event that a BAR packet is received, to report back by using the ACK packet to indicate whether a particular V2X broadcast packet was received (ACK) or not (NACK).

Figure 37 shows the timing and sequence in the transmission of each of the packets. It should be noted that the BAR packet follows the V2X broadcast packet in accessing the medium and that there is no specific timing between them as it is shown in Figure 37, i.e. they both follow the MAC back-off procedures to access the medium that are specified at the MAC LOW layer (see Figure 36). On the other hand, the time between the BAR packet and the ACK packet transmitted by the addressed RxId vehicle is SIFS (Short Inter-Frame Space) as depicted in Figure 37. Note that this is the regular operation at the MAC layer between unicast and ACK packets.

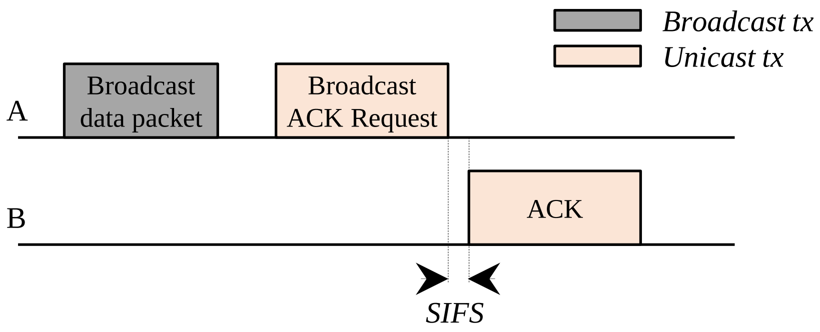


Figure 37: Sequence and timing between the broadcast data packet, the broadcast ACK request packet and the ACK packet.

4.2 Scenario of Evaluation

4.2.1 Simulation environment

This work considers the urban intersection scenario depicted in Figure 38. In the scenario there are vehicles stopped at the traffic light on the horizontal lane, and vehicles approaching to the intersection on the vertical lane. In the scenario, there is a potential risk that the vehicles approaching to the intersection from the vertical lane (e.g. vehicle B) do not detect the pedestrians that are crossing the street using their in-built sensors; e.g. LIDAR sensor that is typically used for pedestrians detection. In the considered scenario, pedestrians crossing the street are only detected by the vehicles that are stopped at the traffic light (e.g. vehicle A). Typically, it would be the first vehicle (vehicle A) since all other vehicles stopped behind would have their front sensors shadowed by the vehicles in front. To improve the awareness of the driving environment, the vehicles in the scenario implement the CPS service over the V2X radio interface. Based on the CPS service, vehicles frequently transmit CPM broadcast messages that include the objects detected by their built-in sensors; the details about the content and the generation of the CPM messages are reported in Section 4.2.2. Detected objects might be either vehicles or pedestrians. Detected pedestrians are represented in Figure 38 by red circles with a bounding box.

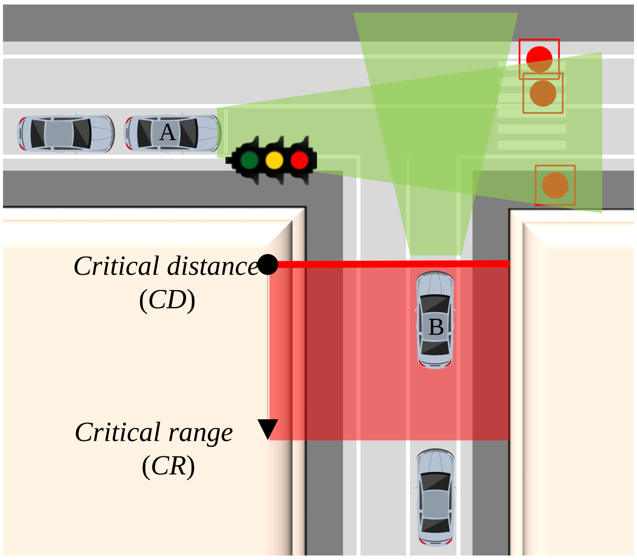


Figure 38: Urban intersection scenario.

The vehicles in the scenario utilize the context-based broadcast ACK mechanism introduced in Section 4 for the CPM messages. In particular, vehicles stopped at the traffic light transmit BAR messages to the vehicles driving on the vertical lane when one of the objects included in their CPM is a pedestrian. In order to identify the vehicle that is addressed in the BAR message, this study has defined a critical distance (*CD*) following the study presented in [60]. The *CD* is defined as the minimum distance to the intersection at which vehicles approaching the intersection on the vertical lane should receive a CPM that includes the pedestrian to avoid a potential collision to the pedestrians at the intersection. Considering a uniform deceleration model, as in [60], the critical distance can be computed as

$$CD = v \cdot RT + v^2/(2 \cdot a_{\max}) \tag{18}$$

where v represents the vehicle's speed, RT the (driver) reaction time, and a_{max} the vehicles' emergency acceleration. Using the *CD* distance as a reference, the vehicle driving on the vertical lane that is closer to the *CD* distance is requested to acknowledge the BAR message. For the sake of scalability of the implemented broadcast acknowledgment mechanism, the proposed implementation limits the request of an ACK packet to the vehicles that are close to the intersection. In this context, this study has also defined a critical range (*CR*) from the *CD* distance to limit the search area of the vehicle that is requested to acknowledge the BAR message.

The studied scenario has been emulated in ns3. The simulator models radio signals propagation losses using the WINNER+ B1 model recommended by 3GPP in [61]. The model implements a log-distance pathloss model that differentiates between LoS and NLoS propagation conditions. It also models the shadowing using a log-normal random distribution with a standard deviation of 3 dB (under LoS) and 4 dB (under NLoS). The vehicles are equipped with IEEE 802.11 radio access technology at 5.9 GHz and transmit CPM messages with a data rate of 6 Mbps using the 1/2 quadrature phase shift keying (QPSK) transmission mode. The transmission power is set to 23 dBm (the antenna gain is 0 dBi). The vehicles driving on the vertical lane move at 20 m/s, and the traffic density is 50 veh/km. Additional simulation parameters are summarized in Table 17.

Table 17: Simulation parameters

Variable	Value
Scenario	
Vehicles speed on the vertical lane (*v*)	20 m/s
Traffic density on the vertical lane	50 veh/km
Critical Range (*CR*)	40 m
Reaction time (*RT*)	{0.75, 1, 1.25} s
Vehicles' emergency acceleration	4 m/s^2
V2X communications	
V2X radio technology	IEEE 802.11p
Antenna gain	0 dBi
Data rate	6 Mbps (QPSK ½)
Frequency/Channel bandwidth	5.9 GHz/10 MHz
Noise figure	9 dB
Energy detection threshold	-85 dBm
Data rate	6 Mbps

4.2.2 CPM's content and generation rules

The CPM messages are transmitted in the studied scenario following the ETSI CPS service's format and generation rules [62]. CPM messages include, among other, Perceived Object Containers (POCs) of 35 bytes each. POCs are optional and provide information about the detected objects (e.g. the distance between the detected object and the transmitting vehicle), the speed and dimensions of the object, and the time at which these measurements were taken.

CPM generation rules define how often a vehicle should generate a CPM and what information is to be included in the CPM. Current ETSI CPM generation rules [62] states that a vehicle has to check every *T_GenCpm* if a new CPM should be generated and transmitted. For our analysis, *T_GenCpm* is set equal to the default 100ms. For every *T_GenCpm*, a vehicle should generate a new CPM if it has detected a new object. For previous detected objects, the vehicle should generate a CPM if any of the following conditions are satisfied:

- For object class pedestrians (in general, Vulnerable Road Users - VRU) or animals:
    - The last time any of the VRU or animal was included in a CPM was 0.5 (or more) seconds ago, include all VRU and animal in a CPM.
- For object class different to VRU or animal:
    - Its absolute position has changed by more than 4m since the last time its data was included in a CPM.

- o Its absolute speed has changed by more than 0.5m/s since the last time its data was included in a CPM.
- o Its absolute velocity has changed by more than 4º since the last time its data was included in a CPM.
- o The last time the detected object was included in a CPM was 1 (or more) seconds ago.

A vehicle includes in a new CPM all new detected objects and those objects that satisfy at least one of the previous conditions. The vehicle still generates a CPM every second even if none of the detected objects satisfy any of the previous conditions. In addition, the information about the onboard sensors is included in the CPM only once per second.

4.3 Results

This section analyses the performance of the proposed context-based broadcast acknowledgment mechanism in the scenario and under the conditions reported in Section 4.2. The proposed mechanism, introduced in Section 4, has been configured to utilize up to 3 retransmissions (i.e. *CounterReTx* = 3) when the transmitting vehicle does not receive the ACK packet. The evaluation also includes a reference technique as a baseline that follows the regular operation of the V2X CPS service, that does not implement the proposed context-based broadcast acknowledgment mechanism, and that therefore does not perform retransmissions of the CPMs (i.e. *CounterReTx* = 0).

In the scenario under evaluation, it is of particular interest to analyse the percentage of vehicles approaching the intersection in the Critical Range that receive at least one CPM, with information about the pedestrian(s). This percentage is shown in Figure 39 as a function of the *Critical Distance*. It should be noted that the different values of the *Critical Distance* can be associated to the driver Reaction Time (RT) using formula (18). In particular, and following [63], the different values that have been considered in this study range from RT=0.0s to RT=1.75s[1]. RT=1.75s can be considered a standard reaction time value [63]. This reaction time reduces in situations where the driver is aware that a dangerous situation might arise (e.g. approaching an intersection). This reaction time is also expected to reduce in vehicles that are controlled by an automated driving software that are designed to electronically react upon risky driving situations. Then, reaction times lower than 1.75 seconds have been also analysed.

The obtained results show, for example, that when the Critical Distance is set to 60 meters to the intersection (i.e. RT=0.5s) the regular V2X CPS service would successfully inform on average 60.8% of the vehicles driving on the vertical lane about the presence of the pedestrians crossing the street. As it is shown in Figure 38, the vehicles driving on the vertical lane are not able to detect the crossing pedestrians using their built-in sensors. Therefore, ~40% of those vehicles are not aware of the presence of the crossing pedestrians with whom they could potentially collide. As the Critical Range moves away from the intersection (i.e. the RT increases), the percentage of vehicles that correctly receive a CPM decreases due to the worse propagation conditions.

When the Critical Distance is set to 85m (i.e. RT=1.75s), only 16% of the vehicles would be aware of the presence of the pedestrians crossing the street without the proposal. Figure 39 shows that when the vehicles implement the proposed context-based broadcast acknowledgment mechanism a higher percentage of vehicles receives at least a CPM within the Critical Range. This is due to the

[1] The Critical Distance 45m is an unfeasible value that would result in a RT = -25s. It has been added for comparative purposes only.

higher reliability in the delivery of the CPM messages. For example, when RT=1.75s, the percentage of vehicles that receives at least a CPM increases to 70.4%, and it reaches ~90% for any value of the RT lower than 0.75s (i.e. Critical Distance lower than 65m).

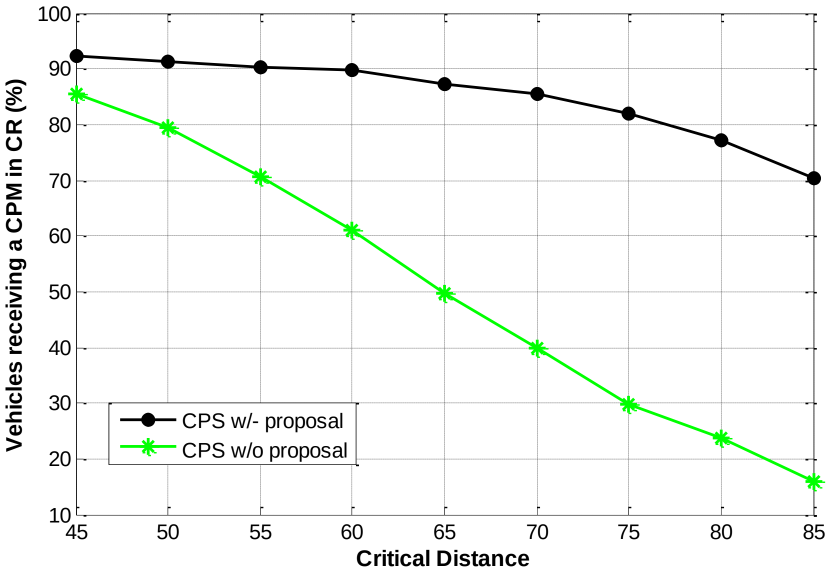


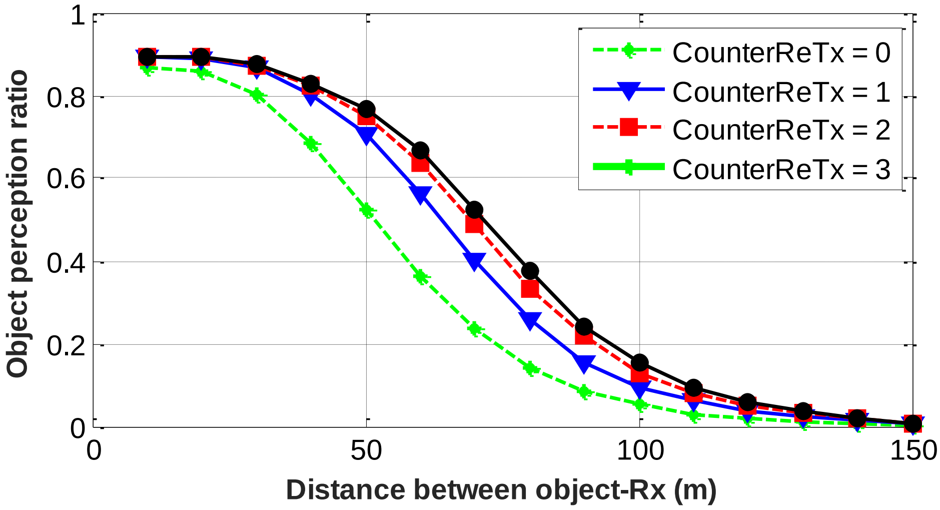
Figure 39: Average percentage of vehicles that receive a CPM within the Critical Range (CR) as a function of the Critical Distance.

Besides the analysis of the awareness of the crossing pedestrians in the Critical Range, this work has also studied the Object Perception Ratio as a function of the distance between the object and the receiving vehicles. The Object Perception Ratio is defined as the probability to detect an object (a crossing pedestrian in this study) through the reception of a CPM with its information in a given time window. Considering the CPM generation rules presented in Section 4.2.2, a CPM with information about the crossing pedestrians is generated every 500ms. Therefore, the time window to calculate the Object Perception Ratio is set to 500ms in this study. In this context, we consider that a crossing pedestrian is successfully detected by a vehicle if it receives at least one CPM with information about that pedestrian per 500ms. This analysis has been conducted considering Critical Distance[2] = {45, 65, 85}m. In addition, different configuration of the proposed mechanism has been analysed in order to assess the impact of the number of retransmissions in the reliability of the delivery of the CPM messages. In particular, the proposed mechanism has been configured to utilize up to *CounterReTx*={1, 2, 3} retransmissions when the transmitting vehicle does not receive the ACK packet. The analysis is also conducted when the vehicles implement the regular V2X CPS service (i.e. *CounterReTx* = 0). Figure 40 depicts the average Object Perception Ratio as a function of the distance between the detected pedestrian and the vehicles receiving the CPM.

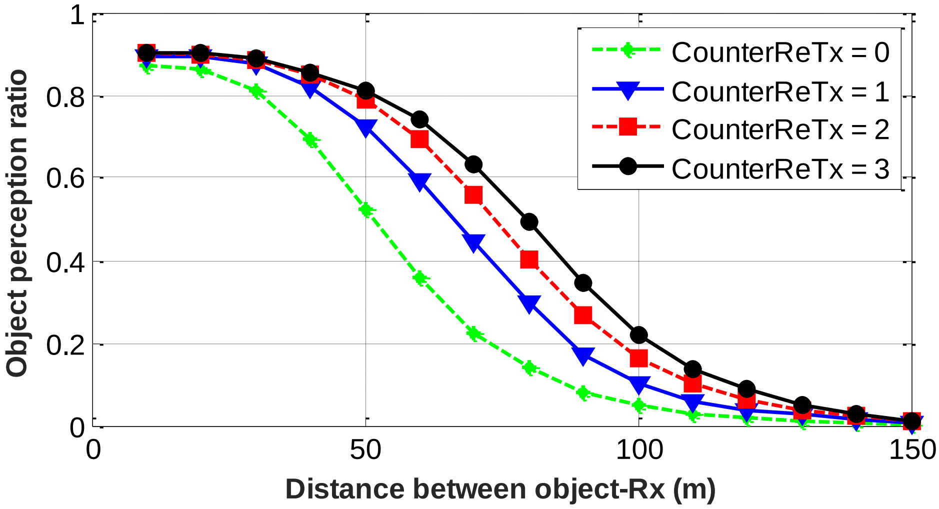
In this study we focus on the vehicles approaching the intersection from the vertical lane. For example, the results reported in Figure 40 show that when the CPS service does not implement the proposed mechanism, the probability that a vehicle detects a crossing pedestrian that is 50 m away through a received CPM is ~50%. This probability decreases when the distance between the vehicles receiving the CPM and the crossing pedestrian increases. The CPM messages with information about the crossing pedestrians are transmitted by the vehicles stopped at the traffic light that are under NLoS conditions to the vehicles approaching the intersection from the vertical lane

[2] Similar trends are shown in this case for this metric a function of the Critical Distance.

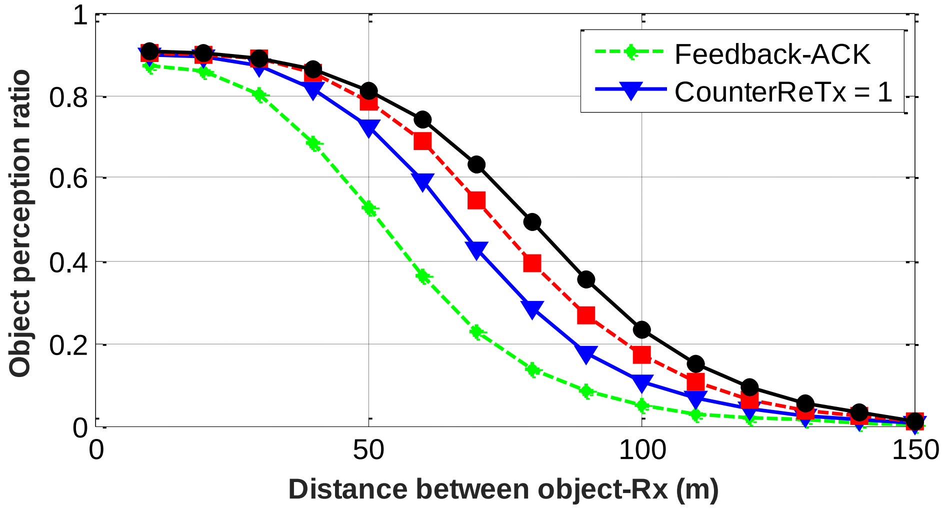
(see Figure 38). The distance between the crossing pedestrians and the vehicles approaching the intersection that need to receive the CPM message is similar to the distance between the vehicles stopped at the traffic light and the vehicles approaching the intersection. Therefore, the worse propagation effects with the increasing distances between the vehicles stopped at the traffic light and the vehicles approaching the intersection result in a reduction of the Object Perception Ratio. The results reported in Figure 40 show that when the vehicles implement the proposed mechanism, the probability to detect the crossing pedestrians increases. For example, when the proposed mechanism is configured to perform up to 3 retransmissions of the CPM (i.e. *CounterReTx*=3), the Object Perception Ratio increases to ~82% when the distance between the vehicle and the crossing pedestrian is 50m. This significant increase demonstrates the potential of the proposed context-based broadcast acknowledgment mechanism to improve the reliability in the delivery of V2X broadcast messages when applied to CPMs.



a) Critical Distance = 45m



b) Critical Distance = 65m



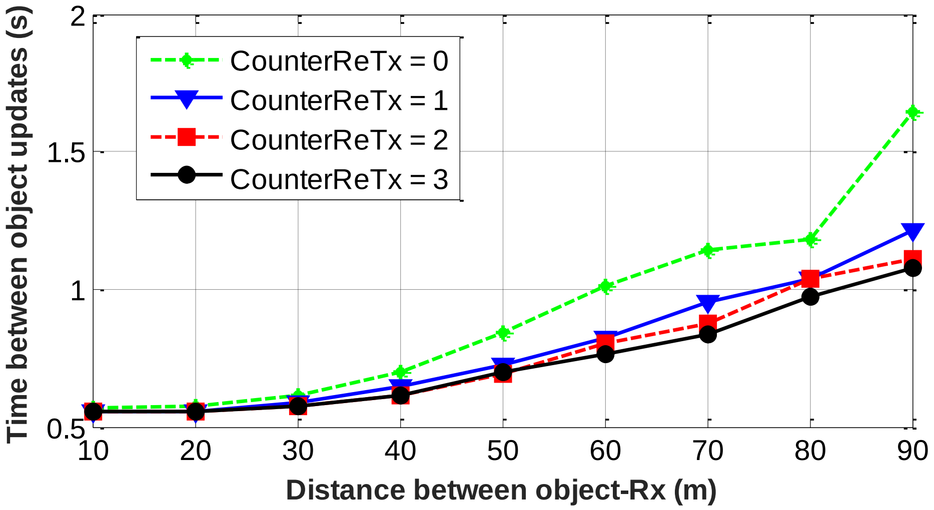
c) Critical Distance = 85m

Figure 40: Object perception ratio as a function of the distance between the object (crossing pedestrian) and the vehicle receiving the CPM. The results are depicted for three different scenarios with Critical Distance = {45, 65, 85} m.

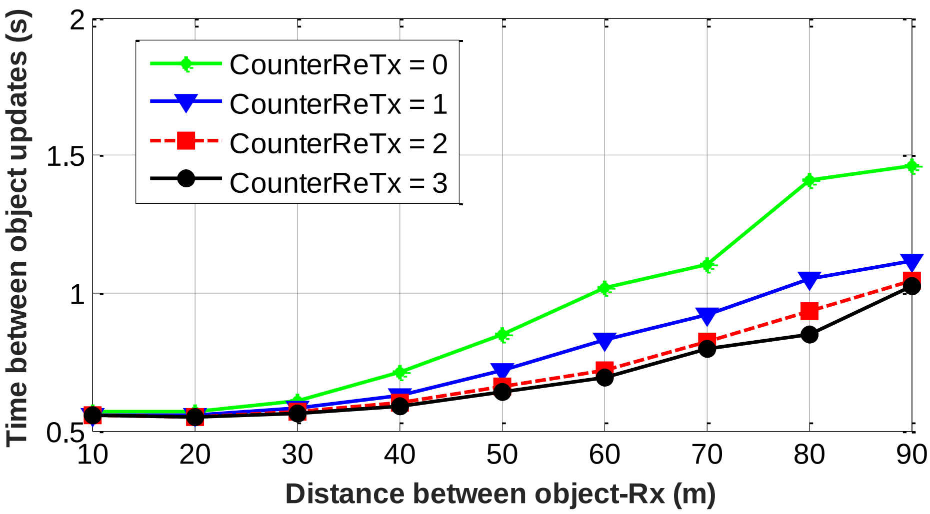
Another important metric for the evaluation of the performance of the CPS, and in particular for the assessment of the awareness of the driving environment supported by the CPS, is the time between object updates. As it has been detailed in Section 4.2.2, CPMs including the crossing pedestrians are generated in the scenario under evaluation every 0.5s. Therefore, vehicles approaching the intersection from the vertical lane would receive a CPM that includes the crossing pedestrian (object) every 0.5s under ideal conditions. However, the results reported in Figure 41 show that the time between updates increases with the distance between the object and the receiver (as indicated above, the distance to the receiver is similar from the object and the transmitting vehicle). For example, when the CPS service does not implement the proposed mechanism, vehicles that are 90 m from the object (crossing pedestrians) receive updates every ~1.5s. This means that vehicles that do not implement the proposed mechanism do not receive two CPM messages for every three CPM messages transmitted.

The results also show that when the vehicles implement the proposed context-based broadcast acknowledgment mechanism they receive more often updates about the crossing pedestrians. For example, in the scenario where the Critical Distance is set to 85m, the vehicles approaching the intersection from the vertical lane receive updates every second when the distance between the vehicle and the object is 90m. This is the case when the proposed mechanism is configured with *CounterReTx*=3. This is the case because of the higher reliability in the delivery of the CPM messages when increasing *CounterReTx because then* the same CPM is retransmitted more times when the addressed vehicle is not responding with the ACK packet.

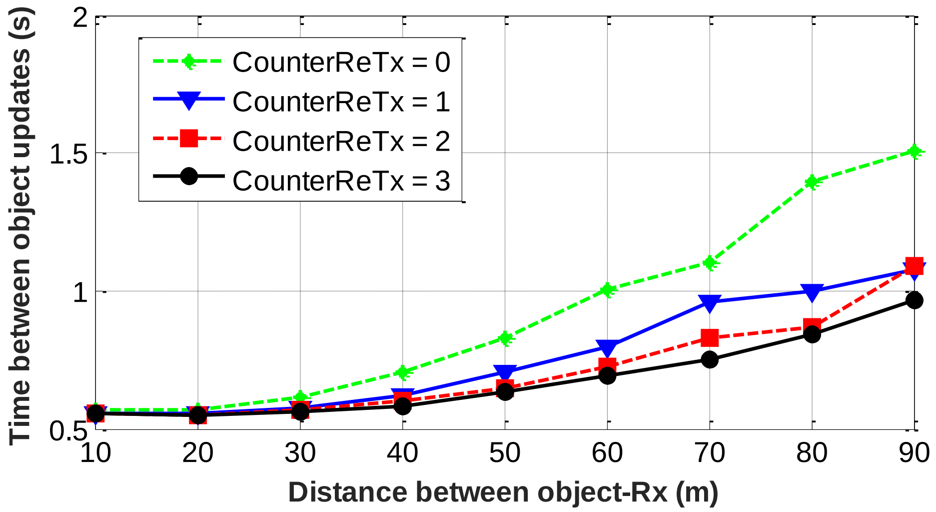
The proposed mechanism has shown to increase the Object Perception Ratio (Figure 40) and to reduce the time between object updates (Figure 41) thanks to the increased reliability in the delivery of the CPM messages. This is achieved by means of the context-based broadcast acknowledgment mechanism that performs a retransmission of the last transmitted CPM if the receiver indicates that the CPM was not received correctly via the ACK packet, or if the timer set to receive the ACK expiries. However, it might also happen that the packet utilized to request the ACK (i.e. BAR packet), or the ACK itself, are not correctly received. If this is the case, the transmitting vehicle would perform a retransmission of a CPM that could have been correctly received, thereby generating a duplicate CPM message at the receiving vehicle. Figure 42 depicts the average number of duplicate CPM messages that are received by the vehicles approaching the intersection, and that include information about the crossing pedestrians, as a function of the distance between the object and the receiver. As it can be observed in Figure 42, the regular V2X CPS service (i.e. *CounterReTx*=0) does not generate any duplicate CPM messages. However, with the proposed mechanism (i.e. *CounterReTx*={1, 2, 3}) there is a certain probability that vehicles receive the same CPM multiple times. This effect is especially high at short distances, because all CPM transmissions and retransmissions are broadcast. Therefore, CPM retransmissions targeted to vehicles in the Critical Range are correctly received with high probability by vehicles that are close to the intersection (simply because they are closer and therefore propagation losses are lower). Figure 42 also shows how the proposed mechanism generates more duplicates of the same CPM when the Critical Distance is set further away to the intersection. In this case, the worse propagation conditions result in that the vehicles generating the CPM require more retransmissions of the CPM in order to inform the vehicles in the Critical Range about the presence of the crossing pedestrians. This side-effect is necessary to increase the probability of receiving a CPM in the Critical Range. The efficiency of the proposed mechanism at the system level is ensured by controlling the number of vehicles and situations where an ACK is requested.



a) Critical Distance = 45

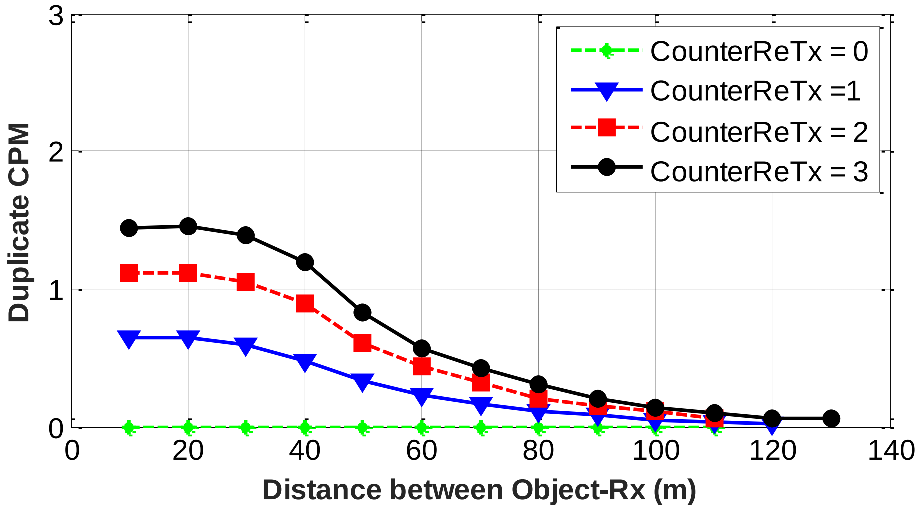


b) Critical Distance = 65

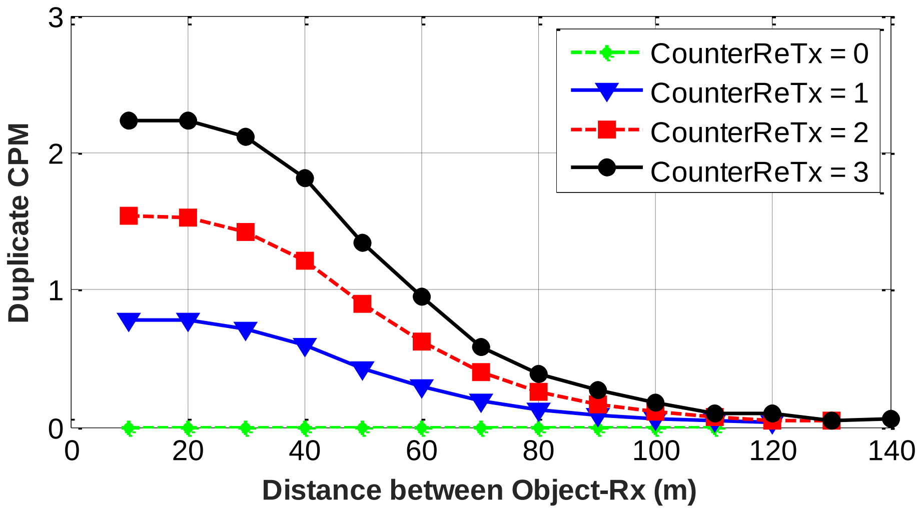


c) Critical Distance = 85

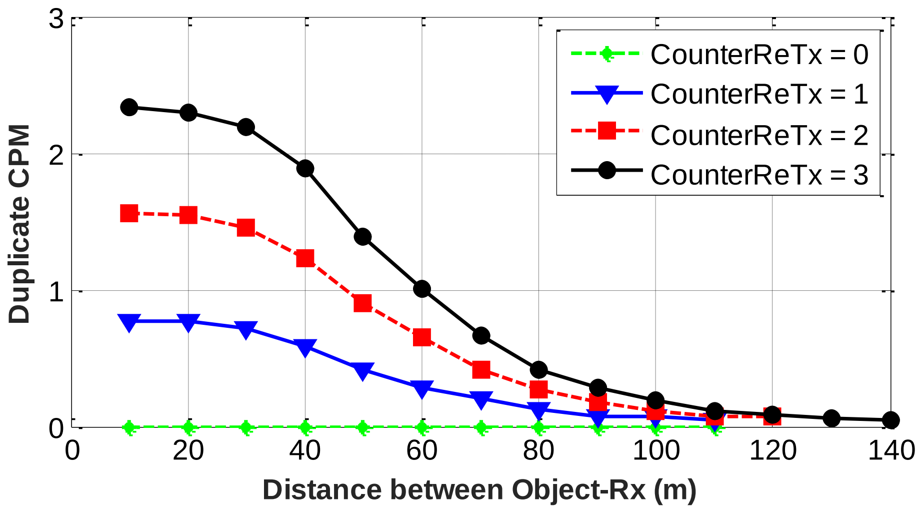
Figure 41: Time between objects updates as a function of the distance between the object (crossing pedestrian) and the vehicle receiving the CPM. The results are depicted for three different scenarios with Critical Distance = {45, 65, 85} m.



a) Critical Distance = 45m



b) Critical Distance = 65m



c) Critical Distance = 85m

Figure 42: Average received duplicate CPM messages as a function of the distance between the object (crossing pedestrians) and the vehicles receiving the CPM. The results are depicted for three different scenarios with Critical Distance = {45, 65, 85} m.

5 Conclusion

The TransAID project has presented in this deliverable the efforts conducted for the design of novel mechanisms to improve the reliability in the exchange of V2X messages between connected/automated vehicles and the infrastructure. TransAID has addressed this issue at different layers of the V2X protocol stack while maintaining their complementarity.

To reduce the negative impact of interferences on the reliability of the V2X message exchange, TransAID has proposed and evaluated the benefits of V2X message compression. Using real-world CAM, CPM and MCM messages, the conducted study has demonstrated that compression gains up to around 50% can be achieved, but this performance significantly depends on the message structure, size and compression method. The obtained results have been used to derive V2X message compression models. These models have been used in network simulations to quantify the effect of compression on the channel load and reliability. The conducted study shows that the channel load can be reduced up to around 27% and the Packet Delivery Ratio can be nearly doubled in some scenarios thanks to data compression. The conducted study has therefore demonstrated the potential of data compression to improve the reliability of V2X communications.

Congestion control is strongly linked with the V2X message exchange reliability, and we have evaluated the ETSI's DCC approaches at Access and Facilities layer, highlighting the trade-off between the radio and application level performance. The results obtained show that ETSI DCC Access can improve the performance at the radio level, but can negatively impact the application level performance. Moreover, its long queuing delays results in the transmission of outdated information. The integration of the Facilities DCC shows that the information age can be significantly improved and that the requirements of different services can be better balanced than with only the Access DCC. TransAID has proposed a novel Facilities DCC that outperforms the one defined in the current ETSI draft. All these solutions that make use of DCC have been compared with TransAID's proposal in deliverable 5.2, demonstrating the potential of intelligent solutions that adapt the amount of information generated by the applications, instead of dropping messages to control congestion.

The reliability of V2X broadcast transmissions is challenged due to the lack of mechanisms to assure the correct delivery of messages. To improve the reliability of V2X broadcast transmissions, TransAID has proposed and evaluated a context-based broadcast acknowledgement mechanism. At the transmitting vehicle, the proposed mechanism selectively requests the acknowledgment of specific/critical V2X broadcast messages and performs retransmissions at the MAC level if they are not correctly received. To improve the scalability of the proposed mechanism, the V2X applications/services are in charge of identifying the situations/conditions that trigger the execution of the broadcast acknowledgment mechanism, and the receiver(s) that should acknowledge the broadcast messages. The proposed mechanism is valid for all types of V2X broadcast messages (e.g. CAMs, DENMs, CPMs, MCMs, etc.) and this Deliverable demonstrates its high potential using the Collective Perception Service. In particular, the reported results have shown that vehicles implementing the proposed mechanism improve their awareness of the driving environment when they are in a critical area with no visibility to pedestrians crossing the street and therefore potentially risk a collision. The proposed mechanism is aligned with the discussions taking place at the IEEE P802.11-Task Group BD and is therefore relevant for the standardization process of IEEE 802.11bd.

References

[1] "Hyundai and Aurora will develop level 4 autonomous vehicles by 2021", Online news: https://www.hyundai.news/eu/technology/hyundai-motor-and-aurora-partner-to-develop-level-4-autonomous-vehicles-by-2021/?L=0

[2] "The new Audi A8 – conditional automated at level 3", Online news: https://www.audi-mediacenter.com/en/on-autopilot-into-the-future-the-audi-vision-of-autonomous-driving-9305/the-new-audi-a8-conditional-automated-at-level-3-9307

[3] "Looking Further - Ford will have a fully autonomous vehicle in operation by 2021", Online news: https://corporate.ford.com/innovation/autonomous-2021.html

[4] "The path to autonomous driving", Online news: https://www.bmw.com/en/automotive-life/autonomous-driving.html

[5] "Autonomous vehicles", Online news: https://group.renault.com/en/innovation-2/autonomous-vehicle/

[6] TransAID, "Use cases and safety and efficiency metrics", Public Deliverable D2.1, March 2018. Online: https://www.transaid.eu/deliverables/

[7] Lu, Z., & De Winter, J. C. F. "A Review and Framework of Control Authority Transitions in Automated Driving" International Conference on Applied Human Factors and Ergonomics, p.p. 2510-2517, 2015.

[8] A. Eriksson, N. A. Stanton, "Driving Performance After Self-Regulated Control Transitions in Highly Automated Vehicles", Human Factors, vol. 59, pp. 1233-1248, 2017

[9] HAVEit, Deliverable D11.1, "Function description and requirements", Sept. 2008

[10] https://www.c-roads.eu/platform/about/news/News/entry/show/c-its-cooperation-between-c2c-cc-and-c-roads-platform-1.html

[11] TransAID, "Scenario definitions and modelling requirements", Public Deliverable D2.2, May 2018. Online: https://www.transaid.eu/deliverables/

[12] TransAID, "System Prototype Demonstration", Public Deliverable D7.2, June 2019. Online: https://www.transaid.eu/deliverables/

[13] ETSI TC ITS, "Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part", TS 102 687 v1.2.1, ETSI, April 2018

[14] IETF, "Hypertext Transfer Protocol -- HTTP/1.1", The Internet Society (1999). Online available: https://tools.ietf.org/html/rfc2616

[15] Andrew B. King, "Speed Up Your Site: Web Site Optimization", New Riders Press, ISBN-10: 0735713243, January 2003.

[16] M. Nelson and J.-L. Gailly, "The Data Compression Book", 2nd edition, Wiley, Dec. 1995.

[17] D.J.C. MacKay, "Information Theory, Inference, and Learning Algorithms", *Cambridge University Press*, 2003.

[18] K. Wesołowski, "Introduction to Digital Communication Systems", Whiley, 2009.

[19] Ubuntu Documentation: Compress. Online [last access on March 2020]: http://manpages.ubuntu.com/manpages/xenial/man1/compress.1.html

[20] GNU Documentation: Gzip. Online [last access on March 2020]: https://www.gnu.org/software/gzip/

[21] J. Ziv and A. Lempel, "A Universal Algorithm for Sequential Data Compression", *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, May 1977.

[22] IETF, "GZIP file format specification version 4.3", The Internet Society (1996). Online available: https://tools.ietf.org/html/rfc1952

[23] Terry A. Welch, "A Technique for High Performance Data Compression", IEEE Computer, vol. 17, no. 6, pp. 8-19, June 1984.

[24] TransAID, "Definition of V2X message sets", Public Deliverable D5.1, August 2019. Online: https://www.transaid.eu/deliverables/
[25] ETSI TC ITS, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service", ETSI EN 302 637-2 V1.3.2, Nov. 2014.
[26] ETSI TC ITS, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS)", ETSI TR 103 562 V2.1.1, Dec. 2019.
[27] 3rd Generation Partnership Project (3GPP), Technical Specification Group Radio Access Network; "Study on LTE-based V2X Services,". TR 36.885 V14.0.0, 2016
[28] M. Sepulcre and J. Gozalvez, "Coordination of Congestion and Awareness Control in Vehicular Networks", Electronics, vol. 7(11), 335, 2018
[29] ETSI, "Intelligent Transport Systems (ITS); Radiocommunications equipment operating in the 5 855 MHz to 5 925 MHz frequency band; Harmonised Standard covering the essential requirements of article 3.2 of Directive 2014/53/EU", ETSI EN 302 571, V2.1.1, Feb. 2017
[30] ETSI, "Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part", ETSI TS 102 687 V1.2.1, April 2018
[31] N. Lyamin, A. Vinel, D. Smely and B. Bellalta, "ETSI DCC: Decentralized Congestion Control in C-ITS," IEEE Communications Magazine, vol. 56, no. 12, pp. 112-118, December 2018
[32] A. Alonso Gómez et al., "Dependability of Decentralized Congestion Control for Varying VANET Density," IEEE Transactions on Vehicular Technology, vol. 65, no. 11, pp. 9153-9167, Nov. 2016
[33] G. Bansal et al., "LIMERIC: A Linear Message Rate Control Algorithm for DSRC Congestion Control," IEEE Transactions on Vehicular Technology, vol. 62 (9), pp. 4182-4197, July 2013
[34] A. Rostami et al., "Stability Challenges and Enhancements for Vehicular Channel Congestion Control Approaches," IEEE Trans. on Intelligent Transportation Systems, vol. 17, no. 10, pp. 2935-2948, Oct. 2016
[35] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5", ETSI TS 102 636-4-2, V1.1.1, Oct. 2013.
[36] ETSI, "Intelligent Transport Systems (ITS); Facilities Layer; Communication Congestion Control", ETSI TS 103 141 V0.0.9, Oct. 2017
[37] M. I. Khan, S. Sesia and J. Harri, "In Vehicle Resource Orchestration for Multi-V2X Services", *Proc. IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, Honolulu, Hawaii, USA, 22–25 September 2019
[38] M. I. Khan and J. Härri, "Integration challenges of facilities-layer DCC for heterogeneous V2X services," *Proc. 29th IEEE Intelligent Vehicles Symposium* (IV), Changshu, Suzhou, China, 26-29 June 2018
[39] ETSI, "Intelligent Transport Systems (ITS); Cross layer DCC management entity for operation in the ITS G5A and ITS G5B medium", ETSI TS 103 175, V1.1.1, June 2015
[40] ETSI, "Intelligent Transport System (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective -Perception Service (CPS) ", TR 103 562 v2.1.1, Dec. 2019.
[41] TransAID, "V2X-based cooperative sensing and driving in Transition Areas", Public Deliverable D5.*2*, Feb. 2019. Online: https://www.transaid.eu/deliverables/

[42] ETSI ITS, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service", EN 302 637-2 V1.3.2, 2014.
[43] 3rd Generation Partnership Project (3GPP), Technical Specification Group Radio Access Network; "Study on LTE-based V2X Services,". TR 36.885 V14.0.0, 2016.
[44] Caltrans Performance Measurement System (PeMS), California Department of Transportation data-set, U.S.State. Available at [last accessed on 2019-01-16]: http://pems.dot.ca.gov/.
[45] ETSI, "Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part", TS 102 687 v1.2.1, April 2018.
[46] ETSI, "Intelligent Transport System (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective -Perception Service (CPS) ", TR 103 562 v2.1.1, Dec. 2019.
[47] A. Correa, et al., "V2X-based cooperative sensing and driving in Transition Areas", *TransAID Deliverable D5.2*, Feb. 2019.
[48] M. Sepulcre and J. Gozalvez, "Coordination of Congestion and Awareness Control in Vehicular Networks", *Electronics*, vol. 7, no. 11, 335, Dec. 2018.
[49] A. Rostami et al., "Stability Challenges and Enhancements for Vehicular Channel Congestion Control Approaches," *IEEE Trans. on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2935-2948, Oct. 2016.
[50] Car 2 Car Communication Consortium, "Guidance for day 2 and beyond roadmap", White paper, September 2019. Available online at [Last accessed in January 2020]: https://www.car-2-car.org/documents/general-documents/.
[51] "In the Matter of: Use of the 5.850-5.925 GHz Band, ET Docket No. 19-138, NOTICE OF PROPOSED RULEMAKING", Released: December 17, 2019, FCC 19-129
[52] 3GPP TS 38.213 V16.0.0, "3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR; Physical layer procedures for control, (Release 16)", December 2019.
[53] Michael Fischer, et al., "Adaptive Repetition", IEEE 802.11-19/1055r0, July 2019.
[54] G. Thandavarayan, M. Sepulcre and J. Gozalvez, "Analysis of Message Generation Rules for Collective Perception in Connected and Automated Driving0", Proc. IEEE Intelligent Vehicle Symposium, Paris (France), June 2019.
[55] Onn Haran, et al., "Reliable V2X operation", IEEE 802.11-19/0717r1, May 2019.
[56] Onn Haran, et al., "Mechanisms for reliable V2X operation", IEEE 802.11-19/1158r2, September 2019.
[57] IEEE 802.11-Task Group BD' official web site [Last accessed in January 2020]: http://www.ieee802.org/11/Reports/tgbd_update.htm.
[58] Onn Haran, et al., "Broadcast Ack Operation", IEEE 802.11-19/2115r1, January 2020.
[59] ETSI ITS, "Intelligent Transport Systems (ITS); Communications Architecture", ETSI EN 302 665 V1.1.1, September 2010.
[60] M. Sepulcre and J. Gozalvez, "On the Importance of Application Requirements in Cooperative Vehicular Communications", 2011 Eighth International Conference on Wireless On-Demand Network Systems and Services, pp. 124-131, Bardonecchia (Italy), 26-28 January 2011.
[61] 3GPP TR 36.885, "Study on LTE-Based V2X services," Rel-14 V14.0.0, July 2016
[62] ETSI ITS, "Intelligent Transport System (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective -Perception Service (CPS) ", ETSI TR. 103 562 V2.1.1, December 2019.

[63] M. Green, “How Long Does It Take To Stop? Methodological Analysis of Driver Perception-Brake Times", Transportation Human Factors, 2000.