



**HAL**  
open science

# Augmented Perception Resiliency of C-ITS based Connected and Autonomous Vehicles

Imed Ghnaya

► **To cite this version:**

Imed Ghnaya. Augmented Perception Resiliency of C-ITS based Connected and Autonomous Vehicles. Other [cs.OH]. Université de Bordeaux, 2024. English. NNT : 2024BORD0068 . tel-04573043

**HAL Id: tel-04573043**

**<https://theses.hal.science/tel-04573043>**

Submitted on 13 May 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE PRÉSENTÉE  
POUR OBTENIR LE GRADE DE

**DOCTEUR DE  
L'UNIVERSITÉ DE BORDEAUX**

ÉCOLE DOCTORALE MATHÉMATIQUES ET INFORMATIQUE  
SPÉCIALITÉ : INFORMATIQUE

Opérée à l'Institut National Polytechnique de Bordeaux (INP Bordeaux)

Par Imed GHNAYA

**Résilience de la Perception Collective et Augmentée  
des Véhicules Autonomes Connectés par les C-ITS**

**Sous la direction de : Mohamed MOSBAH**

**Co-directeurs : Toufik AHMED et Marion BERBINEAU**

**Soutenue le : 17/04/2024**

**Membres du jury :**

M. SENOUCI, Sidi Mohammed	PR, Université de Bourgogne, Nevers	Rapporteur
M. SONDI, Patrick	PR, IMT Nord Europe, Lille	Rapporteur
M. BRADAI, Abbas	PR, Université de Côte d'Azur, Nice	Président
M. MOSBAH, Mohamed	PR, UB, Bordeaux INP, Talence	Directeur de thèse
M. AHMED, Toufik	PR, UB, Bordeaux INP, Talence	Co-directeur de thèse
Mme. BERBINEAU, Marion	DR, Université Gustave Eiffel, Villeneuve-d'Ascq	Co-directrice de thèse
Mme. ANISS, Hasnaa	IR, Université Gustave Eiffel, Talence	Encadrante

# Acknowledgements

I would like to express my deepest gratitude to my thesis supervisors, Mohamed Mosbah, Toufik Ahmed, Hasnaa Aniss, and Marion Berbineau, for their invaluable guidance and support throughout the period of this research. Their expertise, insightful suggestions, and encouragement have been instrumental in shaping both the direction and execution of this work, even in the toughest moments.

I am also immensely grateful to the members of the jury, Sidi Mohammed Senouci, Patrick Sondi, and Abbas Bradai, whose constructive feedback and assessments have significantly contributed to the refinement and depth of this work. Their perspectives and critiques have been crucial in enhancing the quality of my thesis.

My sincere appreciation also goes to the entire team and my colleagues at LaBRI and COSYS/ERENA for providing a stimulating and supportive environment for my research. The discussions, collaborations, and shared knowledge within this community have been a constant source of inspiration and motivation.

Finally, I would like to extend my sincere gratitude and deep love to my parents and sister. Your unwavering support, endless patience, and encouragement have been my guide throughout this journey.

THANK YOU ALL

# Abstract

Cooperative Intelligent Transport Systems (C-ITS) are pivotal in driving the future mobility towards more autonomous, efficient, and safer roads. These systems leverage communication technologies, such as European Telecommunications Standards Institute (ETSI) ITS-G5 and Cellular Vehicle-to-Everything (C-V2X), to enhance road safety and traffic flow. Central to C-ITS is the concept of Cooperative Perception (CP), enabling Cooperative and Autonomous Vehicles (CAVs) to share data gathered by onboard sensors like cameras, lidar, and radar with nearby CAVs. This cooperation is facilitated through the exchange of Cooperative Perception Messages (CPMs) via V2X communication.

Building upon the concept of CP, Cooperative and Augmented Perception (CAP) significantly enhances the perceptual capabilities of CAVs. A key development in CAP is its integration within infrastructure, equipping traffic entities such as Road-Side Units (RSUs), traffic lights, and road signs with advanced sensors. These sensors not only offer broaden perception capabilities but also share additional perceptual information with CAVs, facilitating an advanced cooperative data fusion process. This process, supported by edge and cloud computing technologies, enables a more comprehensive understanding of the driving environment.

The primary goal of CAP is to boost environmental awareness about road objects for CAVs, particularly in complex environments and under various network conditions. However, realizing the full potential of CAP and ensuring its continuous operation face significant challenges. A major challenge lies in managing and filtering the increasing volume of perception data to share via V2X. This data, often containing inaccurate, irrelevant, and redundant information, can overwhelm the communication network and lead to an increased congestion. Addressing these challenges require efficient data filtering strategies that can not only manage the data volume but also to fulfill the specific needs of receiving CAVs for relevant information and dynamically adapt to the varying complexity of the driving environment and changing network congestion levels and control mechanisms.

Motivated by the need to overcome these challenges, this thesis focuses on enhancing the resilience of CAP from networking and communication perspectives, leveraging recent advancements in modeling and Artificial Intelligence (AI). Specifically, it leverages the power of Deep Reinforcement Learning (DRL) for adaptive behaviors across various complex scenarios with fluctuating conditions.

The contributions made in this thesis are as follows. First, we propose a Vehicle-to-Vehicle Redundancy Mitigation Method (V2V-RMM) that minimizes redundant data transmission between CAVs while maintaining perception awareness. This approach employs Deep Q-Network (DQN) and Double DQN (DDQN) learning principles in a multi-agent setting. Second, we introduce a Context-Aware V2X Cooperative and Augmented Perception approach (MDRL-CR) using the Advantage Actor-Critic (A2C) algorithm in a distributed setting,

enabling CAVs and RSUs to learn relevance-aware and context-sensitive strategies for transmitting perception data, tailored to the specific needs of each CAV and the current challenges it faces. Next, we introduce Distributed Congestion Control (DCC)-Aware Cooperative and Augmented Perception (MDRL-CR-DCC), leveraging Asynchronous A2C (A3C) learning for tackling the negative impact of congestion control mechanisms on CAP quality. These contributions are validated through simulations using OMNeT++, Artery, and Simulation of Urban Mobility (SUMO). Evaluation results demonstrate the effectiveness of the proposed contributions in enhancing perception awareness for CAVs over safety-critical distances of less than 200 meters, reducing latency, and improving CAP's resilience in environments with varied conditions and under network limited resources.

# Résumé étendu en français

## Contexte et Motivation

Le développement des systèmes avancés d'aide à la conduite (ADAS) constitue une étape cruciale vers la réalisation de la conduite autonome, visant à renforcer les capacités de conduite humaine pour améliorer la sécurité routière et l'efficacité opérationnelle grâce à l'intégration de capteurs variés comme le radar, le lidar et les caméras. Toutefois, malgré des progrès notables, les ADAS rencontrent des limites, notamment la restriction liée à la ligne de visée des capteurs, entravée par des obstacles physiques tels que d'autres véhicules, des bâtiments ou la complexité de la topographie routière. Les communications véhicule-à-tout (V2X) offrent une solution à ces limitations, reposant sur les technologies ITS-G5 de l'ETSI pour les communications courtes et la norme C-V2X du 3GPP pour les interactions avec des noeuds distants comme le cloud. Dans ce contexte, la perception coopérative (CP) est introduite comme une solution essentielle, permettant aux véhicules autonomes et connectés (CAVs) de partager les données de perception avec d'autres CAVs via la communication V2X. Ce principe améliore significativement la sécurité routière en élargissant le champ de perception limité de chaque véhicule au-delà de sa propre ligne de visée. Ce concept de CP peut être augmenté par l'intégration de la perception au niveau de l'infrastructure, appelé CP augmentée (CAP), permettant à des composants tels que les unités de bord de route (RSUs) et les feux de circulation d'être équipés de capteurs avancés et de transmettre des informations de perception additionnelles aux CAVs.

## Problématique et Contributions

Cependant, exploiter pleinement le potentiel de la CAP et garantir sa continuité opérationnelle risquent de relever plusieurs défis. Un principal défi est de gérer l'augmentation du volume de données de perception générées par les capteurs embarqués des CAVs, des RSUs et reçues via les communications V2X. Ces données, potentiellement imprécises, non pertinentes ou redondantes, risquent de saturer les réseaux de communication et de causer une congestion accrue. Cela met en lumière l'importance de développer des stratégies efficaces de filtrage de données. Ces dernières doivent traiter de manière efficiente le volume important de données et s'adapter dynamiquement aux complexités de l'environnement de conduite, répondre aux besoins spécifiques des CAVs récepteurs en informations pertinentes, et s'ajuster aux variations des conditions de congestion du réseau et aux mécanismes du contrôle. Pour assurer un système de CAP résilient et robuste, il est crucial que les CAVs reçoivent des données fiables et pertinentes pour une navigation sécurisée, même en cas de défaillances du réseau et d'interférences évolutives.

Motivée par la nécessité de surmonter ces défis, cette thèse vise à améliorer la résilience de la CAP, particulièrement du point de vue de la communication et du réseau. Cet effort est appuyé par des avancées récentes dans le domaine de la modélisation et de l'intelligence artificielle, notamment par l'utilisation de l'apprentissage profond par renforcement (DRL) pour adapter les comportements dans divers scénarios et conditions fluctuantes. Les contributions apportées par cette thèse, validées à travers des simulations réalisées avec les simulateurs OMNeT++, Artery et Simulation de Mobilité Urbaine (SUMO), incluent :

**Méthode de Mitigation de Redondance des Objets (V2V-RMM) :** Cette méthode consiste à introduire une stratégie avancée basée sur l'apprentissage multi-agents par DRL (MDRL) pour réduire la transmission de données de perception redondantes entre les CAVs. Dans un environnement congestionné avec une densité de véhicules élevée, le même objet peut être détecté par plusieurs véhicules, menant à des transmissions redondantes qui surchargent le réseau et diminuent la perception de l'environnement. V2V-RMM exploite des algorithmes DRL, tels que Deep Q-Network (DQN) et Double DQN (DDQN), pour optimiser le processus de sélection et de transmission d'objets via les messages CPMs, en privilégiant la transmission d'informations utiles tenant compte l'évaluation de différents paramètres comme la distance, la taille, l'angle de vue et les obstructions potentielles. Les résultats des simulations démontrent que cette méthode réduit efficacement la redondance des objets et augmente la fiabilité du réseau, améliorant ainsi la conscience d'objets dans l'environnement sur des distances courtes et moyennes inférieures à 200 mètres par rapport aux méthodes existantes.

**MDRL pour une Perception Coopérative et Augmentée V2X Sensible au Contexte (MDRL-CR) :** Cette contribution propose une approche d'apprentissage innovante par Advantage Actor-Critic (A2C) pour assurer la résilience de la CAP dans des contextes variés et complexes. MDRL-CR permet aux CAVs et aux RSUs d'apprendre des stratégies de filtrage et de transmission de données de perception via les CPMs, sensibles au contexte et à la pertinence. Cette méthode, ajustée au contexte spécifique de chaque CAV récepteur, a montré son efficacité en améliorant significativement la transmission d'informations cruciales pour la sécurité dans des conditions de trafic denses et complexes, réduisant le taux d'occupation du canal de 20 à 40 % et diminuant la latence de communication d'environ 35 % par rapport aux méthodes existantes.

Apprentissage par Renforcement Asynchrone pour une Perception Coopérative et Augmentée Sensible à la Congestion et Adaptée au DCC (MDRL-CR-DCC): Cette contribution présente MDRL-CR-DCC afin de répondre aux défis posés par les mécanismes de contrôle de la congestion et leur impact négatif sur la qualité de la CAP. Cette méthode, qui intègre un apprentissage asynchrone adapté aux modalités opérationnelles du DCC, optimise la transmission des CPM en fonction des différents niveaux de congestion. Les résultats démontrent que la sélection adéquate d'objets pertinents, ajustant ainsi la taille des CPMs au seuil fixé par le DCC, réduit efficacement la congestion du réseau. Ceci améliore la fiabilité du réseau et, par conséquent, la résilience de la CAP face aux variations environnementales et aux restrictions imposées par les contraintes du DCC.

# Contents

ABSTRACT .....	III
RÉSUMÉ ÉTENDU EN FRANÇAIS .....	III
ACRONYMS .....	III
LIST OF SYMBOLS .....	IIII
PUBLICATIONS .....	XV
1 INTRODUCTION .....	1
1.1 BACKGROUND AND CONTEXT .....	1
1.2 MOTIVATIONS AND PROBLEM STATEMENT .....	2
1.3 RESEARCH OBJECTIVE AND CONTRIBUTIONS .....	3
1.4 THESIS ORGANISATION AND OUTLINE .....	4
2 THE EVOLUTION OF COOPERATIVE INTELLIGENT TRANSPORTATION SYSTEMS: A STATE OF THE ART .....	5
2.1 INTRODUCTION TO C-ITS .....	5
2.1.1 DEFINITION AND BENEFITS .....	5
2.1.2 C-ITS STATIONS .....	5
2.1.3 COMMUNICATION SCENARIOS .....	6
2.2 C-ITS REFERENCE ARCHITECTURE AND STANDARDS .....	7
2.2 ETSI ITS-S REFERENCE ARCHITECTURE AND STANDARDS .....	8
2.2.1 COMMUNICATION TECHNOLOGIES .....	9
2.2.2 NETWORKING AND TRANSPORT PROTOCOLS .....	13
2.2.3 MANAGEMENT AND SECURITY ENTITIES .....	15
2.2.4 C-ITS SERVICES AND RELATED MESSAGES .....	16
2.2.5 DECENTRALIZED CONGESTION CONTROL .....	20
2.2.6 C-ITS APPLICATIONS .....	23
2.3 PARADIGMS AND INNOVATION IN C-ITS .....	24
2.3.1 EMERGING PARADIGMS .....	24
2.3.2 TECHNOLOGICAL INNOVATIONS IN C-ITS .....	25
2.4 CONCLUSION AND FUTURE DIRECTIONS OF C-ITS .....	26
3 COOPERATIVE AND AUGMENTED PERCEPTION IN C-ITS: AN OVERVIEW .....	27
3.1 INTRODUCTION .....	27
3.2 AUTONOMOUS VEHICLES AND PERCEPTION .....	27
3.2.1 OVERVIEW ON AUTONOMOUS VEHICLES .....	27
3.2.2 RECENT ADVANCES IN SENSING TECHNOLOGIES .....	29
3.2.3 MULTISENSORY LOCAL DATA FUSION .....	31
3.2.4 LIMITATIONS OF INDIVIDUAL AV PERCEPTION SYSTEM .....	32
3.3 COOPERATIVE AND AUGMENTED PERCEPTION .....	33



3.3.1 DEFINITION AND BENEFITS OF CP .....	33
3.3.2 STANDARDIZATION EFFORTS .....	35
3.3.3 SIMULATION TOOLS.....	36
3.3.4 AUGMENTED CP .....	37
3.4 RESILIENCE OF THE COOPERATIVE AND AUGMENTED PERCEPTION .....	41
3.4.1 DEFINITION AND KEY CONCEPTS .....	41
3.4.2 AREA OF CHALLENGES AND RESILIENCY REQUIREMENTS .....	41
3.5 CONCLUSION.....	42
4 MULTIAGENT DEEP REINFORCEMENT LEARNING FOR OBJECT REDUNDANCY MITIGATION IN V2V NETWORKS .....	43
4.1 INTRODUCTION .....	43
4.2 RELATED WORKS .....	44
4.2.1 ETSI RMMS .....	44
4.2.2 FURTHER ENHANCED RMMS.....	45
4.3 PROBLEM FORMULATION .....	46
4.3.1 ARCHITECTURE.....	47
4.3.2 OPTIMIZATION PROBLEM.....	48
4.4 SYSTEM MODEL AND LEARNING ALGORITHMS .....	50
4.4.1 MOTIVATION ON DRL .....	50
4.4.2 SYSTEM MODEL.....	51
4.4.3 LEARNING ALGORITHMS .....	52
4.5 PERFORMANCE EVALUATION.....	57
4.5.1 SCENARIO AND PARAMETERS.....	58
4.5.2 ANALYSIS OF TRAINING CONVERGENCE.....	58
4.5.3 IMPACT ON NETWORK PERFORMANCE AND CAP QUALITY .....	59
4.6 CONCLUSION.....	61
5 ADVANTAGE ACTOR-CRITIC LEARNING FOR CONTEXT-AWARE V2X COOPERATIVE AND AUGMENTED PERCEPTION.....	63
5.1 INTRODUCTION .....	63
5.2 RELATED WORKS .....	64
5.2.1 DISTANCE-BASED METHODS.....	64
5.2.2 DL-BASED METHODS .....	65
5.2.3 RL-BASED METHODS .....	65
5.3 ARCHITECTURE AND MODELING .....	66
5.3.1 SYSTEM COMPONENTS AND COMMUNICATION.....	66
5.3.2 FUNCTIONAL ARCHITECTURE.....	67
5.3.3 CONTEXT-AWARE MODELLING.....	69
5.4 LEARNING ALGORITHM FOR CONTEXT-AWARE CAP.....	71
5.4.1 MODELLING OF THE MULTI-AGENT RL ENVIRONMENT .....	71
5.4.2 ADVANTAGE ACTOR-CRITIC LEARNING .....	73
5.5 SIMULATION RESULTS AND EVALUATIONS.....	75
5.5.1 SCENARIO AND COMMUNICATION PARAMETERS .....	75
5.5.2 EVALUATION OF THE TRAINING PHASE: REWARD AND LOSS VARIATION .....	76

5.5.3 EVALUATION OF THE EXPLOITATION PHASE: IMPACT ON COOPERATIVE PERCEPTION AND NETWORK RELIABILITY .....	78
5.6 CONCLUSION.....	84
6 ASYNCHRONOUS RL FOR CONGESTION-AWARE COOPERATIVE PERCEPTION IN VEHICULAR NETWORKS ENABLED BY DCC .....	86
6.1 INTRODUCTION .....	86
6.2 IMPACT OF DCC ON CAP QUALITY.....	87
6.3 CONGESTION-AWARE PROBLEM FORMULATION AND MODELING.....	88
6.3.1 PROBLEM FORMULATION .....	88
6.2.1 SYSTEM MODEL.....	89
6.3 ASYNCHRONOUS RL FOR CONGESTION-AWARE COOPERATIVE PERCEPTION .....	91
6.3.1 ASYNCHRONOUS RL IN DCC-ENABLED VEHICULAR NETWORKS .....	91
6.3.1.1 MOTIVATION AND CHALLENGES .....	91
6.3.2 MDRL-CR-DCC LEARNING ALGORITHMS.....	92
6.4 EVALUATION RESULTS .....	97
6.4.1 REWARD AND LOSS VARIATION.....	99
6.4.2 EVALUATION OF CAP QUALITY.....	100
6.5 CONCLUSION.....	102
7 CONCLUSION AND FUTURE RESEARCH .....	104
7.1 CONCLUSION.....	104
7.2 LIMITATIONS AND PERSPECTIVES.....	105
7.2.1 SHORT-TERM GOALS .....	105
7.2.2 LONG-TERM PERSPECTIVES .....	106

# Table of Figures

FIGURE 2.1 THE C-ITS ECOSYSTEM AND ITS NETWORK ARCHITECTURE [4] .....	6
FIGURE 2.1 ITS-S REFERENCE ARCHITECTURE .....	8
FIGURE 2.2 ITS-G5 CHANNELS .....	10
FIGURE 2.3 LTE-V2X ARCHITECTURE [12] .....	12
FIGURE 2.4 GEO-NETWORKING ROUTING SCHEMES [16] .....	14
FIGURE 2.5 GENERAL STRUCTURE OF A CAM [30] .....	17
FIGURE 2.6 GENERAL STRUCTURE OF A DENM [33] .....	18
FIGURE 2.7 GENERAL STRUCTURE OF A VAM [34] .....	19
FIGURE 2.8 DCC ARCHITECTURE [39] .....	21
FIGURE 2.9 REACTIVE DCC STATE MACHINE [41] .....	22
FIGURE 3.1 EXAMPLE OF AN AV .....	28
FIGURE 3.2 STANDARD AUTOMATION LEVELS [47] .....	29
FIGURE 3.3 SENSORS OF AN AUTONOMOUS VEHICLES [48] .....	30
FIGURE 3.4 THE CONCEPT OF CP IN AN URBAN AREA .....	34
FIGURE 3.5 BASIC ARCHITECTURE OF CP .....	34
FIGURE 3.6 CPM FORMAT [74] .....	36
FIGURE 3.7 COOPERATIVE AND AUGMENTED PERCEPTION IN C-ITS .....	39
FIGURE 3.8 ARCHITECTURE OF COOPERATIVE AND AUGMENTED PERCEPTION IN CAVs .....	40
FIGURE 3.9 ARCHITECTURE OF COOPERATIVE AND AUGMENTED PERCEPTION IN INFRASTRUCTURE .....	40
FIGURE 4.1 BASIC SERVICES USED BY EACH CAV .....	47
FIGURE 4.2 AN ILLUSTRATION OF ROAD OCCLUSIONS IN THE DRIVING ENVIRONMENT .....	48
FIGURE 4.3 DQN BASIC COMPONENTS INTERNAL THE CAV .....	54
FIGURE 4.4 THE AVERAGE REWARD VARIATION OF V2V-RMM USING MDQN AND MDDQN, AS A FUNCTION OF TRAINING EPISODES. ....	58
FIGURE 4.5 OBJECT REDUNDANCY AS A FUNCTION OF THE DISTANCE BETWEEN THE PERCEIVED OBJECT AND THE CAV RECEIVING IT. ....	60
FIGURE 4.6 CPA AS A FUNCTION OF THE DISTANCE BETWEEN THE PERCEIVED OBJECT AND THE CAV RECEIVING IT. ....	61
FIGURE 4.7 CPM DELIVERY RATIO AS A FUNCTION OF THE DISTANCE TRANSMITTER-RECEIVER	61
FIGURE 5.1 NETWORK ARCHITECTURE WITHIN A HETEROGENEOUS ENVIRONMENT OF CONNECTED AND AUTONOMOUS DRIVING .....	67
FIGURE 5.2 ARCHITECTURE OF MDRL-CR .....	68
FIGURE 5.3 AVERAGE REWARD VARIATION OF MDRL-CR .....	77

FIGURE 5.4 AVERAGE LOSS VARIATION OF MDRL-CR .....	77
FIGURE 5.5 AVERAGE REWARD VARIATION OF MCORM USING A2C ALGORITHM, AS A FUNCTION OF TRAINING EPISODES. ....	79
FIGURE 5.6 AVERAGE LOSS VARIATION OF MCORM USING A2C ALGORITHM, AS A FUNCTION OF TRAINING EPISODES. ....	79
FIGURE 5.7 VARIATION OF THE AVERAGE NUMBER OF VRU AND DISCONNECTED OBJECTED RECEIVED BY CAVs AS A FUNCTION OF THEIR DCS.....	81
FIGURE 5.8 VARIATION OF THE AVERAGE SPEED OF VRU AND DISCONNECTED OBJECTED RECEIVED BY CAVs AS A FUNCTION OF THEIR DCS.....	81
FIGURE 5.9 OBJECT PERCEPTION RATIO AS A FUNCTION OF THE DISTANCE BETWEEN THE DETECTED OBJECT AND THE CAV RECEIVING IT.....	82
FIGURE 5.10 THE VARIATION OF CBR IN CCH AS A FUNCTION OF SIMULATION TIME INTERVALS. ....	83
FIGURE 6.1 POSSIBLE FOV OF A CAV .....	90
FIGURE 6.2 ILLUSTRATION OF THE A3C PRINCIPLE [124].....	93
FIGURE 6.3 ARCHITECTURE OF MDRL-CR-DCC BASED ON A3C.....	94
FIGURE 6.4 AVERAGE REWARD VARIATION AS A FUNCTION OF TRAINING EPISODES OF THE GLOBAL MODEL.....	99
FIGURE 6.5 AVERAGE LOSS VARIATION AS A FUNCTION OF TRAINING EPISODES OF THE GLOBAL MODEL OF A3C.....	100
FIGURE 6.6 CPM SIZE.....	101
FIGURE 6.7 THE CBR AS A FUNCTION OF SIMULATION TIME INTERVAL.....	101
FIGURE 6.8 THE CP AWARENESS AS A FUNCTION OF THE DISTANCE BETWEEN THE DETECTED OBJECT AND THE RECEIVING CAV .....	102

# Table of Tables

TABLE 2.1 TRANSFER RATES AND CODING RATES IN 802.11P [1], [2].....	10
.....	10
TABLE 2.2 AIFS AND CW SIZES FOR 802.11P's ACs [1], [2] .....	11
TABLE 2.3 REACTIVE DCC PARAMETER LOOK-UP TABLE .....	23
TABLE 5.1 PARAMETERS USED FOR SIMULATION OF MDRL-CR.....	76
TABLE 5.2 LATENCY OF CPMs IN CCH .....	84
TABLE 6.1: PARAMETERS USED FOR SIMULATION OF MDRL-CR-DC.....	98

# Acronyms

3GPP	Generation Partnership Project
A2C	Advantage Actor-Critic
A3C	Asynchronous Advantage Actor-Critic
AC	Actor-Critic
ADAS	Advanced Driver Assistance Systems
AV	Autonomous Vehicle
BEV	Bird Eye View
CA	Cooperative Awareness
CAM	Cooperative Awareness Message
CAP	Cooperative and Augmented Perception
CAS	CA Service
CAV	Connected and Autonomous Vehicle
CBR	Channel Busy Ratio
CCDA	Centralized Critic for Decentralized Actors
CCH	Control Channel
CDR	CPM Delivery Ratio
CP	Cooperative Perception
CPA	CP Awareness
CPS	CP Service
CSMA/CA	Carrier Sense Multiple Access with Collision
CTDE	Centralized Training with Decentralized Execution
C-V2X	Cellular-V2X
CW	Contention Window
DCC	Decentralized Congestion Control
DDQN	Double Deep Q-Network
DE	Data Element
DEN	Decentralized Environmental Notification
DENM	DEN Message
DFM	Data Fusion Module
DFS	Dynamic Feature Sharing
DL	Deep Learning
DP	DCC Profile
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
DSRC	Dedicated Short-Range Communications
EC	Edge Computing
EDCA	Enhanced Distributed Channel Access
EDM	Environment Descriptor Module

ETSI	European Telecommunications Standards Institute
FoV	Field of View
GEM	Global Environmental Model
GEMO	GEM Object
GN	GeoNetworking Protocol
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
I2P	Infrastructure-to-Pedestrian
I2V	Infrastructure-to-Vehicle
ICT	Information and Communication Technologies
IMP	Information Management and Provider
ITS	Intelligent Transport System
ITS-S	ITS Stations
IVC	Inter-Vehicle Communication
LDM	Local Dynamic Map
LEM	Local Environmental Model
LEMO	LEM Object
Lidar	Light Detection and Ranging
LoS	Line-of-Sight
LPS	Local Perception Service
LTE V2X	LTE-Vehicle to Everything
MAC	Medium Access Control
MC	Management Container
MCO	Multi-Channel Operation
MCS	Modulation and Coding Schemes
MDRL	Multi-Agent Deep Reinforcement Learning
MEC	Multi-access Edge Computing
MIB	Management Information Base
MLP	Multilayer Perceptron
OCB	Outside the Context of a Basic Service Set
OFDM	Orthogonal Frequency Division Multiplexing
OR	Object Redundancy
P2I	Pedestrian-to-Infrastructure
P2V	Pedestrian-to-Vehicle
PDU	Packet Data Unit
POC	Perceived Object Container
R2E	RSU-to-Edge
R2V	RSU-to-Vehicle
Radar	Radio Detection and Ranging
RMM	Redundancy Mitigation Method
RSU	Roadside Units
SCH	Service Channel
SDC	Station Data Container
SIC	Sensor Information Container

SPAT	Signal Phase and Timing
SUMO	Simulation of Urban Mobility
TraCI	Traffic Control Interface
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2P	Vehicle-to-Pedestrian
V2R	Vehicle-to-RSU
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VAM	VRU Awareness Message
Veins	Vehicles in Network Simulation
VRU	Vulnerable Road User



# List of Symbols

$i, j, k$	Indexes
$t_{av}, t_{ar}$	Time instances related to the awareness of CAVs and RSUs
$t_{pv}, t_{pr}$	Time instances related to cooperative perception of CAVs and RSUs
$x, y, sp$	Position (x,y) and speed
$l, w$	Length and width
$o, cl$	A detected object and its class
$T, M$	The set of all perceived and the selected objects to include in the CPM
$P(o)$	The priority of object $o$
$\alpha_1, \dots, \alpha_4$	The possible values of $P(o)$
$SC_v, m_v$	The driving safety coverage of range $m_v$ for CAV $v$
$DCS_v$	The driving context score for CAV $v$
$d_{v,v'}$	The distance between two CAVs $v$ and $v'$
$f, g$	Distance-related and occlusion-related factors
$h$	The status information
$H$	The set of status information received from other CAVs
$L$	CPM length
$\bar{L}$	Estimated CPM length
$\bar{L}'$	Last Estimated CPM length
$freq$	The CPM generation frequency
$N$	Quantifies the impact of the CPM size regarding DCC restrictions
$F_{o,r}$	Factor assessing the capability of a receiver $r$ to observe an object $o$ as a function position, speed, size, and road occlusions
<i>Cells</i>	The matrix-based representation of the field-of-view
$C_{i,j}^k$	A rectangular cell specified by its row $i$ , column $j$ , and indexed by $k$ in <i>Cells</i>
$s, a, r$	The fundamental RL elements: state, action, and reward
$e, \mathcal{D}$	A single experience, a buffer of accumulated experiences
$\theta, \theta^-$	DNN parameters
<i>Loss</i>	The loss of the critic network
$A(s, a)$	The value indicating the advantage of action $a$ when taken in state $s$
$\tau$	A local history of state-action pairs
$n_e$	The size of the sampled minibatch of experiences from $\mathcal{D}$
$\alpha, \gamma$	Learning rate, discount factor

# List of Publications

## Journal Papers

Ghnaya, I., Aniss, H., Ahmed, T., & Mosbah, M. Reliable Perception for Connected and Autonomous Vehicles via Multi-Channel Operations and Deep Reinforcement Learning. Submitted to: *IEEE Transactions on Network and Service Management Journal* (2023)

## Conference Papers

Ghnaya, I., Ahmed, T., Mosbah, M., & Aniss, H. Asynchronous Learning for Congestion-Aware Cooperative Perception in Vehicular Networks Enabled by DCC. Submitted to: *2024 49th IEEE Conference on Local Computer Network* (2024)

Ghnaya, I., Mosbah, M., Aniss, H., & Ahmed, T. (2023, May). Multi-Agent Advantage Actor-Critic Learning For Message Content Selection in Cooperative Perception Networks. In *NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium* (pp. 1-9). IEEE.

Ghnaya, I., Aniss, H., Ahmed, T., & Mosbah, M. (2023, March). A Distributed Double Deep Q-Learning Method for Object Redundancy Mitigation in Vehicular Networks. In *2023 IEEE Wireless Communications and Networking Conference (WCNC)* (pp. 1-6). IEEE.

Ghnaya, I., Ahmed, T., Mosbah, M., & Aniss, H. (2022, October). Maximizing Information Usefulness in Vehicular CP Networks Using Actor-Critic Reinforcement Learning. In *2022 18th International Conference on Network and Service Management (CNSM)* (pp. 296-302). IEEE.

# Chapter 1

## Introduction

### 1.1 Background and Context

Modern transportation is undergoing a significant evolution with the integration of Intelligent Transport Systems (ITSs). These frameworks are pivotal in steering the future of mobility towards more autonomous, efficient, and safer roads. ITSs leverage cutting-edge technologies and data analytics to improve traffic management, reduce congestion, and enhance the driving experience. The advent of ITS is particularly promising for the advancement of autonomous driving technologies, which are set to revolutionize the future of personal and public transport.

A crucial step towards achieving fully Autonomous Vehicles (AVs) is the development of efficient Advanced Driver Assistance Systems (ADAS). These systems represent a synergy of various technologies designed to augment human driving capabilities, thereby increasing road safety and operational efficiency. By utilizing an array of onboard sensors, such as radar, lidar, and cameras, ADAS provide AVs with a comprehensive understanding of their immediate environment. This capability enables them to detect obstacles, monitor traffic conditions, and even recognize traffic signs, facilitating informed decision-making processes.

However, despite remarkable progress in sensor technology, these systems are not without their limitations. One of the most significant challenges facing ADAS today is the inherent restriction of their sensors' line-of-sight (LoS). This limitation, primarily due to the physical constraints of sensor technology, can be hindered by various obstructions, such as other vehicles, buildings, and natural elements like trees and complex road geometry. Moreover, the effectiveness of these sensors can be compromised by adverse weather conditions, including heavy rain, fog, and snow, further exacerbating the challenge of achieving fully autonomous driving. In response to these limitations, the research and development community is actively exploring complementary technologies that can extend the capabilities of current ADAS. Among these technologies, Vehicle-to-Everything (V2X) communication emerges as a particularly promising solution. It marks a paradigm shift in enhancing the situational awareness of Connected AVs (CAVs), facilitating a seamless exchange of relevant and timely information among various traffic participants. This includes not only their interaction with other CAVs in the vicinity but also communication with Roadside Units (RSUs), such as connected road signs and tolls, as well as with pedestrians and other connected traffic entities, significantly improving awareness.

V2X communication is supported by leading standards that ensure interoperability and reliability across different devices and systems. Notably, the European Telecommunications Standards Institute (ETSI) ITS-G5 profile specializes in short-range communications, enabling direct, real-time exchange of information between CAVs and RSUs, through Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) interactions, respectively. On the other hand, the 3rd Generation Partnership Project (3GPP) Cellular-V2X (C-V2X) standard supports both short and extended-range communications. C-V2X is particularly aimed at facilitating Vehicle-to-Network (V2N) communications, connecting CAVs with far-reaching network services, such as cloud computing platforms. These interactions are facilitated through the exchange of Cooperative Awareness Messages (CAMs), representing a primary means of exchanging status information, such as position, speed, and direction via V2X. However, the continuous exchange of self-status between CAVs and RSUs has its limitations, especially in providing a comprehensive perception of the driving environment. Notably, CAMs might not include information about non-V2X-equipped road traffic participants, such as pedestrians, motorcycles, and bicycles.

With onboard sensors also limited in their perception range, there is an increased risk of compromising driving safety and increasing the potential for accidents. To address this issue, Cooperative Perception (CP) is introduced as a fundamental solution, enabling CAVs to share perception data with other CAVs using V2X communication. This concept significantly improves road safety by extending each CAV's limited perceptual horizon beyond the Field of View (FoV) of its onboard sensors, enabling the detection of potential hazards and obstacles, and thus significantly improving road safety.

## **1.2 Motivations and Problem Statement**

Building upon the concept of CP, the introduction of Cooperative and Augmented Perception (CAP) represents a further advancement in enhancing the perceptual capabilities of CAVs. A pivotal aspect introducing CAP is its integration on the infrastructure side, allowing traffic entities such as RSUs, traffic lights, and road signs to be equipped with advanced sensors. These sensors not only possess enhanced capabilities but also relay additional perceptual information to CAVs. This integration supports a sophisticated cooperative data fusion process, where information perceived locally is combined and cross-validated with data received from remote CAVs and infrastructure, enabling more comprehensive understanding of the driving environment. Furthermore, the adoption of edge and cloud computing technologies plays a pivotal role in augmenting the CP of CAVs. Considering the immense volume of data generated by onboard sensors and network cooperation, these technologies are crucial for reducing latency and guaranteeing rapid and timely responses in various dynamic road scenarios.

However, achieving the full potential of CAP and ensuring its operational continuity requires surmounting a series of challenges. A primary challenge involves addressing the increasing volume of data generated by onboard sensors in CAVs, RSUs, and received via V2X communications. This data, potentially comprising imprecise, irrelevant, and redundant information, can saturate communication networks and causes increased congestion. This limitation highlights the necessity for efficient data filtering strategies. Such strategies must not

only efficiently manage the high volume of data but also adapt dynamically to the diverse complexities of the driving environment, meet the specific needs of receiving CAVs for relevant information, and the changing conditions of network congestion and control mechanisms. To ensure a resilient CAP system, it is crucial that CAVs receive timely and relevant data for safe navigation, even under evolving network failures and interference.

## 1.3 Research Objective and Contributions

Motivated by the necessity to overcome the aforementioned challenges, this thesis is dedicated to enhancing the resilience of CAP from networking and communications perspectives. This effort is supported by recent advancements in modeling and Artificial Intelligence (AI), particularly the use of Deep Reinforcement Learning (DRL) to adapt behaviors under various scenarios and fluctuating conditions. The key contributions made by this thesis, validated through simulations performed with the OMNeT++, Artery and Simulation of Urban Mobility (SUMO) simulators, include:

- **V2V Redundancy Mitigation Method (V2V-RMM):** This contribution introduces an advanced strategy utilizing Multi-Agent Deep Reinforcement Learning (MDRL) to mitigate the transmission of redundant perception data between CAVs. In congested environments with increased vehicle density, the same objects might be detected by several CAVs, triggering unnecessary data transmissions that can overload the communication channel and consequently reduce environmental awareness. V2V-RMM leverages MDRL, incorporating principles such as Deep Q-Network (DQN) and Double DQN Network (DDQN), to improve the decision-making process for selecting and disseminating objects through CPMs. This method prioritizes the transmission of relevant information by evaluating various parameters, including distance, size, viewing angle, and potential obstructions in the driving environment. Simulation results highlight that the proposal effectively mitigates object redundancy and improves network reliability, ensuring increased awareness at short and medium distances of less than 200 m compared to state-of-the-art approaches.
- **MDRL for Context-Aware V2X Cooperative and Augmented Perception (MDRL-CR):** This contribution proposes an innovative Advantage Actor-Critic (A2C) learning approach to augment CAP systems in varied and complex environments. MDRL-CR enables CAVs and RSUs to learn relevance-aware and context-sensitive strategies for the filtering and transmission of perception data via CPMs. Tailoring this process to the unique environmental context of each receiving CAV, the approach has demonstrated significant improvements in providing safety-critical information in dense traffic and complex conditions. Furthermore, MDRL-CR enhances CAP quality, notably reducing Channel Busy Ratio (CBR) by 20% to 40%, and decreasing communication latency by approximately 35% compared to existing works.
- **Asynchronous DRL for DCC-Aware Cooperative and Augmented Perception (MDRL-CR-DCC):** Addressing the integration of congestion control mechanisms and their impact on CAP quality, this contribution introduces MDRL-CR-DCC. It incorporates an asynchronous DRL framework to align with the DCC operational mode,

optimizing CPM transmission in response to varying congestion levels. The results demonstrate that selecting the appropriate relevant objects to adjust the size of CPMs to the size threshold imposed by DCC effectively reduced congestion in the channel. This, in turn, improved network reliability and consequently enhanced the CAP's resilience in the varying environment and network conditions imposed by DCC restrictions.

## 1.4 Thesis Organization and Outline

The rest of this thesis is structured as follows:

Chapter 2 explores the foundational elements of C-ITS, detailing its core components, architectural frameworks, and services. This chapter also discusses the integration of key technologies such as machine learning, communication protocols, cloud computing, and edge computing within C-ITS.

Chapter 3 delves into concept of CAP within C-ITS, underscoring its potential to enhance road safety and driving efficiency. It emphasizes the importance of resilience mechanisms for effective operation under various adversities and challenges, particularly in ensuring the reliability of information and communication technologies.

Chapter 4 presents the first contribution, V2V-RMM, showcasing a MDRL-based approach to minimize the exchange of redundant perception data through CPMs CAVs. It elaborates on the methodology, the optimization challenge it addresses, and the significant results achieved through simulations.

Chapter 5 introduces the second contribution, MDRL-CR, which leverages a distributed version of the A2C algorithm within the driving environment. This chapter further enhances the CAP capabilities of CAVs and RSUs, prioritizing data based on the urgency and contextual needs of each receiving vehicle.

Chapter 6 describes the third contribution, MDRL-CR-DCC, employing an asynchronous DRL principle, particularly the Asynchronous A2C (A3C) algorithm, to improve CAP quality in congested networks enabled by DCC.

Chapter 7 concludes this thesis and presents some directions for future work.

# Chapter 2

## The Evolution of Cooperative Intelligent Transportation Systems: A State of the Art

### 2.1 Introduction to C-ITS

The efficiency of transportation systems is crucial for the global economy, enabling the movement of people and goods across various regions. Road transportation, a key component of these systems, faces significant challenges that affect its effectiveness and safety. The grim reality of road safety is marked by approximately 1.35 million annual fatalities and an additional 20-50 million people suffering injuries or disabilities [1]. Furthermore, the traffic congestion, particularly in urban areas, leads to not only to time delays and decreased operational efficiency but also intensifies environmental issues, such as air and noise pollution. These challenges underscore the urgent need for systemic improvements and the integration of innovative technologies in road transportation to achieve safety, efficiency, and sustainability goals.

#### 2.1.1 Definition and Benefits

C-ITS is introduced as a promising solution. Its core development is predicated on the principle of information exchange among all road users to provide road users with real-time and actionable information about the road environment, improving road safety, efficiency. This is done by leveraging wireless communication and data sharing to facilitate dynamic interactions between vehicles and different road elements [2].

#### 2.1.2 C-ITS Stations

The C-ITS ecosystem includes an interconnected network of various ITS Stations (ITS-S), as depicted in Figure 2.1. These stations are categorized into the following distinct types [3]:

- Central ITS-S: These stations provide centralized applications for traffic and road operations, often functioning as cloud servers or data centers that offer a range of services and contents.

- Roadside ITS-S (RSU): Positioned alongside roads, these stations monitor road traffic independently or in tandem with other ITS-Ss. Their roles include transmitting traffic-related information, broadcasting safety warnings, and assessing traffic conditions.
- Vehicle ITS-S: Within vehicles, these stations offer applications to drivers and passengers, accessing and communicating a broad spectrum of in-vehicle data with other ITS stations.
- Personal ITS-S: These stations cater to personal and portable devices, such as smartphones and wearable tech, facilitating individual communication within the C-ITS network.

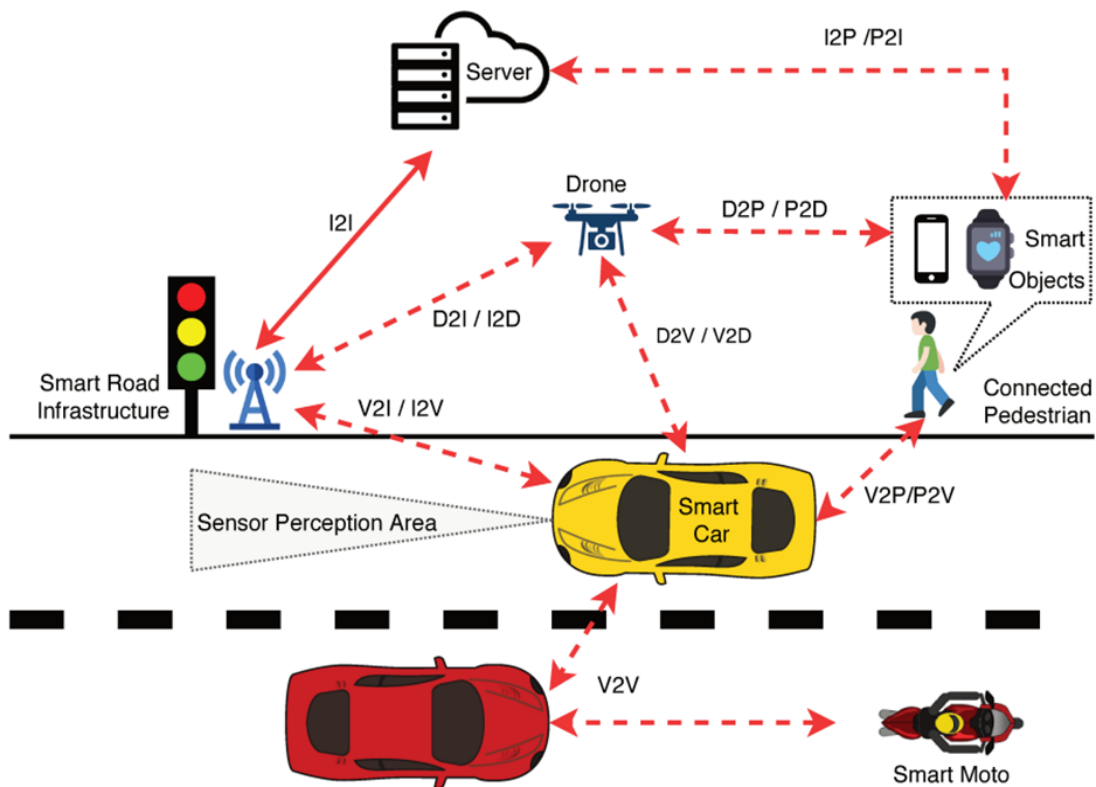


Figure 2.1 The C-ITS ecosystem and its network architecture [4]

### 2.1.3 Communication Scenarios

The interaction between different ITS-S significantly enhances data accuracy and reliability concerning vehicle positioning and road conditions. This exchange is facilitated through various communication scenarios, each contributing uniquely to the C-ITS network:

- V2V Communication: This scenario involves direct information exchange between vehicles, creating an autonomous, ad hoc network independent of fixed infrastructure. The primary purpose of this communication scenario is to exchange necessary information about positioning, traffic congestion, accidents, and speed limits [5] and [6].



- V2I and Infrastructure-to-Vehicle (I2V) Communications: These scenarios cover the diverse interactions between vehicles and different types of infrastructure.
- V2N: Using this mode, vehicles communicate with further distant network entities, such as cloud and Multi-access Edge Computing (MEC) devices accessible over the Internet.
- V2P and Pedestrian-to-Vehicle (P2V) Communications: These interactions bridge the gap between vehicles and pedestrians, thereby potentially augmenting pedestrian safety. By increasing situational awareness, these communications help in alerting both drivers and pedestrians of potential hazards.
- Pedestrian-to-Infrastructure (P2I) and Infrastructure-to-Pedestrian (I2P) Communications: These exchanges involve pedestrians communicating with road infrastructure, such as traffic signals and crosswalks, and vice versa. These interactions play a crucial role in ensuring pedestrian safety and efficient urban mobility.

## 2.2 C-ITS Reference Architecture and Standards

The development of C-ITS globally necessitates the development of various standardization organizations, each tailored to a specific regulatory, technological, and infrastructural needs of different regions. This diversity in standardization bodies lies in the unique challenges and requirements encountered in different parts of the world, such as varying traffic conditions, technological advancements, and regulatory environments. Consequently, a range of organizations has emerged, each contributing to the standardization process in their respective regions.

In Europe, ETSI plays a pivotal role, focusing on Information and Communication Technologies (ICT) standards. These standards are designed to ensure interoperability and reliability within the European context. In the United States, the IEEE and SAE International are worldwide organizations, with IEEE focusing on wireless communication technologies and SAE on automotive-related standards. However, Japan's standardization efforts are led by the Association of Radio Industries and Businesses (ARIB) and ITS Japan, focusing on radio systems and overall promotion of ITS technologies, respectively. Additionally, organizations like ISO and 3GPP also provide overarching frameworks that facilitate international interoperability and technological integration.

This thesis, focusing on the European context, primarily adheres to the standards set by ETSI and 3GPP. These standards are particularly relevant for aligning C-ITS developments and applications with European regulations and technological frameworks, making them suitable for implementation in European C-ITS environments.

# 2.2 ETSI ITS-S Reference Architecture and Standards

ETSI has played a pivotal role in shaping the future of C-ITS through its development of a comprehensive ITS reference architecture, as depicted in Figure 2.1. This architecture serves as a blueprint, outlining the various components and layers that constitute the core of an ITS-S. It is structured to facilitate seamless communication and data exchange among all entities involved in the transportation system, including vehicles, infrastructure, and other road users. This architecture is based on several key layers:

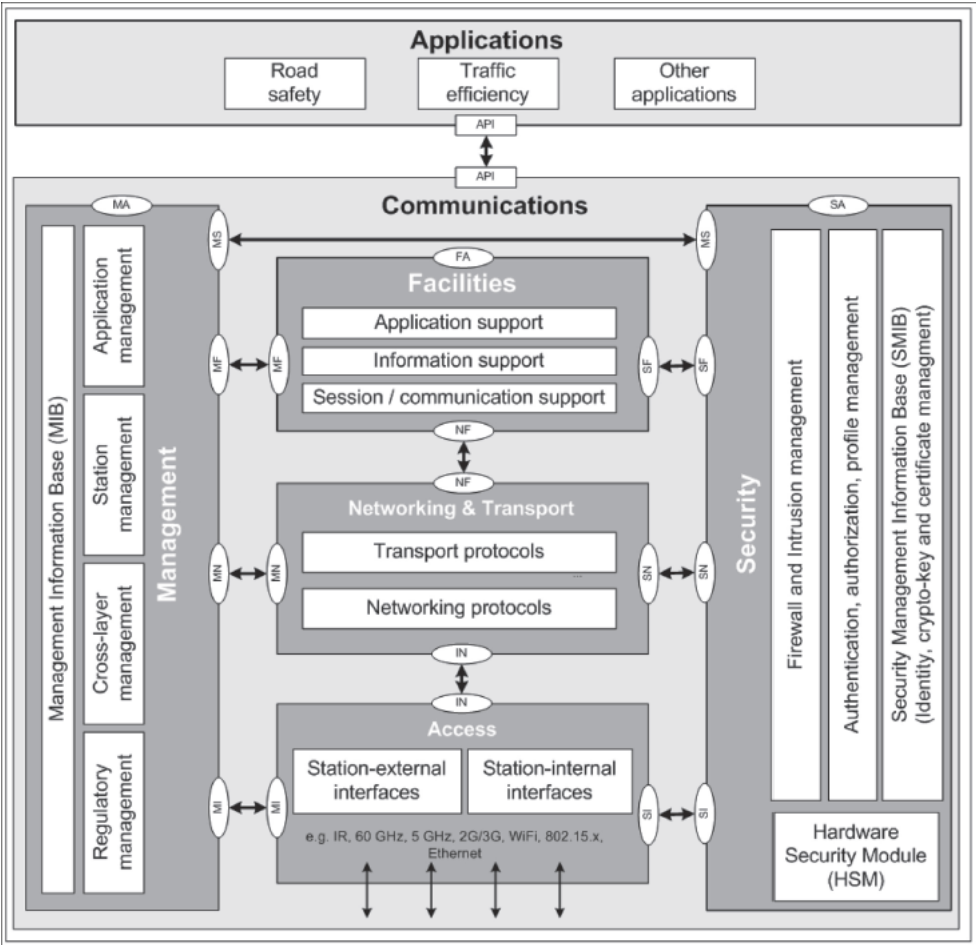


Figure 2.1 ITS-S reference architecture [3]

- Access Layer: This layer composes of two distinct sublayers. The first sublayer incorporates hardware components and sensors that are embedded within the ITS-S. These elements are fundamental for data collection and facilitating the initial interaction with the physical environment. The second sublayer comprises protocols for both the physical and link layers. This sublayer is instrumental in enabling connectivity and facilitating communication across a variety of network types. It supports diverse

communication technologies, including cellular, Wi-Fi, and Dedicated Short-Range Communications (DSRC), thus ensuring efficient data transmission within the C-ITS framework.

- **Networking & Transport Layer:** This layer encompasses the routing and transport protocols for data exchange between ITS-Ss.
- **Facilities Layer:** Handles data interpretation, aggregation, processing, and management. It is crucial for the translation of raw data into actionable information to support ITS applications.
- **Application Layer:** This layer hosts various ITS applications that directly interact with end-users, such as road safety and efficiency related applications.
- **Management Layer:** This layer plays a crucial role in overseeing the entire ITS framework. It involves the administration and coordination of various system components, ensuring efficient operation, resource allocation, and maintenance of the ITS network.
- **Security Layer:** Essential for safeguarding the ITS against potential threats and vulnerabilities. This layer encompasses protocols and measures to protect data integrity, privacy, and system functionality from cyber threats and unauthorized access.

## **2.2.1 Communication Technologies**

This section introduces the basic channel access technologies used for communication between ITS-Ss within the C-ITS framework.

### **2.2.1.1 ITS-G5 Technology**

ITS-G5 represents a channel access technology built upon the IEEE 802.11p standard, mainly designed for V2V and V2I communications. At the physical layer, ITS-G5 leverages Orthogonal Frequency Division Multiplexing (OFDM) with 52 sub-carriers, with 48 allocated for data transmission and 4 designated for pilot signals [7], [8]. Table 2.1 illustrates the resulting data rates for ITS-G5 using different Modulation and Coding Schemes (MCSs), with the 3, 6 and 9 Mbps required for all ITS-S.

ITS-G5 frequencies are allocated depending on their purpose of use, which also differ on performance requirements. To enable various ITS applications, one Control Channel (CCH) and 7 Service Channels (SCH) are allocated, as depicted by Figure 2.2 and detailed in Table 2.1. For the two spectrum bands, ITS-G5A and ITS-G5B, the channel spacing is set at 10 MHz. The OFDM symbol duration is 8  $\mu$ s which includes a cyclic prefix (guard interval) of 1.6  $\mu$ s. The maximum delay spread should ideally not exceed 1.6 $\mu$ s, which corresponds to a distance of about 500 meters with a maximum communication range of less than 1 km [9].

At the Medium Access Control (MAC) layer, ITS-G5 adopts an ad-hoc mode known as Outside the Context of a Basic Service Set (OCB). This mode empowers ITS-S stations to bypass conventional Wi-Fi operations like channel accessing and association, enabling the direct and immediate transmission of messages employing a wildcard Basic Service Set (BSS)

identifier [1]. This configuration is particularly effective for ITS applications that require rapid and reliable V2V and V2I communications, as summarized in Table 2.1, which outlines the data rates and modulation schemes of IEEE 802.11p adopted by ITS-G5.

Data rate (Mbps)	Modulation scheme	Coding rate	Data bits per OFDM symbol	Coded bits per OFDM symbol
3	BPSK	1/2	24	48
4.5	BPSK	3/4	36	48
6	QPSK	1/2	48	96
9	QPSK	3/4	72	96
12	16-QAM	1/2	96	192
18	16-QAM	3/4	144	192
24	64-QAM	2/3	192	288
27	64-QAM	3/4	216	288

Table 2.1 Transfer rates and coding rates in 802.11p [1] and [2]

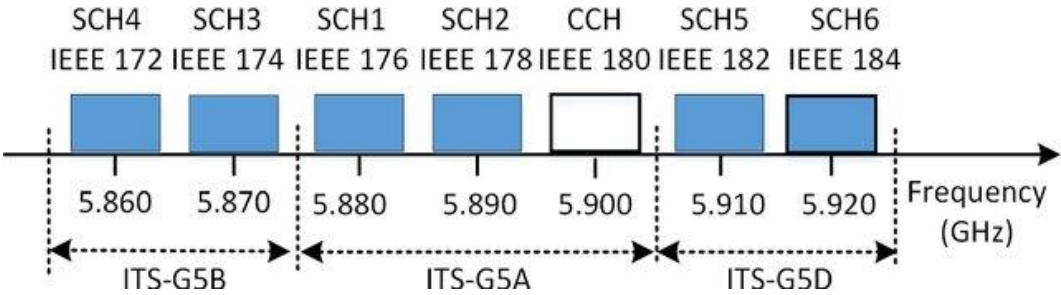


Figure 2.2 ITS-G5 channels [1] and [2]

In this context, the Enhanced Distributed Channel Access (EDCA) mechanism is utilized. EDCA builds upon the basic Distributed Coordination Function (DCF), which employs the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) channel access algorithm. Within the framework of EDCA, four separate queues are established, each with its

unique listening periods, Arbitration Inter-Frame Space (AIFS), and Contention Window (CW) settings, as shown in Table 2.2. Each node maintains multiple queues, each with different AIFS values and CW sizes. This arrangement is designed to increase the likelihood of data traffic with higher priority gaining access to the communication channel before data traffic with lower priority. To facilitate this prioritization, IEEE 802.11p defines 4 User Priorities (UP) that are ranging from the lowest priority “0” to the highest priority 3, as illustrated in Table 2.2.

User Priorities (UP)	Access category (AC)	CW_min	CW_max	AIFS ( $\mu$ s)
1	AC_Voice (VO)	3	7	58
2	AC_Video (VI)	7	15	71
3	AC_Best Effort (BE)	15	1023	110
4	AC_Background (Bk)	15	1023	149

Table 2.2 AIFS and CW sizes for 802.11p’s ACs [1] and [2]

The MAC layer of ITS-G5 also allows for the operation over multiple channels, known as Multi-Channel Operation (MCO) [10]. Briefly, MCO is a fundamental aspect of ITS-G5 technology that empowers ITS-S, such as CAVs and RSUs, to simultaneously utilize multiple communication channels. This capability significantly enhances network capacity, ensures continuous data exchange, and strengthens communication resilience, particularly in challenging or congested traffic environments.

### 2.2.1.2 LTE-V2X (LTE-Uu/PC5)

LTE-Vehicle to Everything (LTE-V2X) technology is also a channel access technology that leverages the existing LTE network infrastructure to facilitate a wide range of communication possibilities, from short-range direct communication to long-range network-based communication between ITS-Ss through V2N [9].

LTE-V2X operates through two main communication radio interfaces: LTE-PC5, also known as LTE side-link, and LTE-Uu, as depicted in Figure 2.3. The LTE-PC5 (Side-Link) allows for direct communication between different ITS-S, such as CAVs, RSUs, and pedestrians, without the need for a cellular network infrastructure, known as eNodeB. It is optimized for safety-critical applications that require low latency and high reliability, using techniques like Time Division Multiplexing (TDM) and Frequency Division Multiplexing (FDM) to handle high vehicle speeds and densities. Meanwhile, the LTE-Uu interface connects ITS-S using the eNodeB. It supports a broader range of services, including cloud-based applications and remote vehicle management, suitable for less time-sensitive communications.

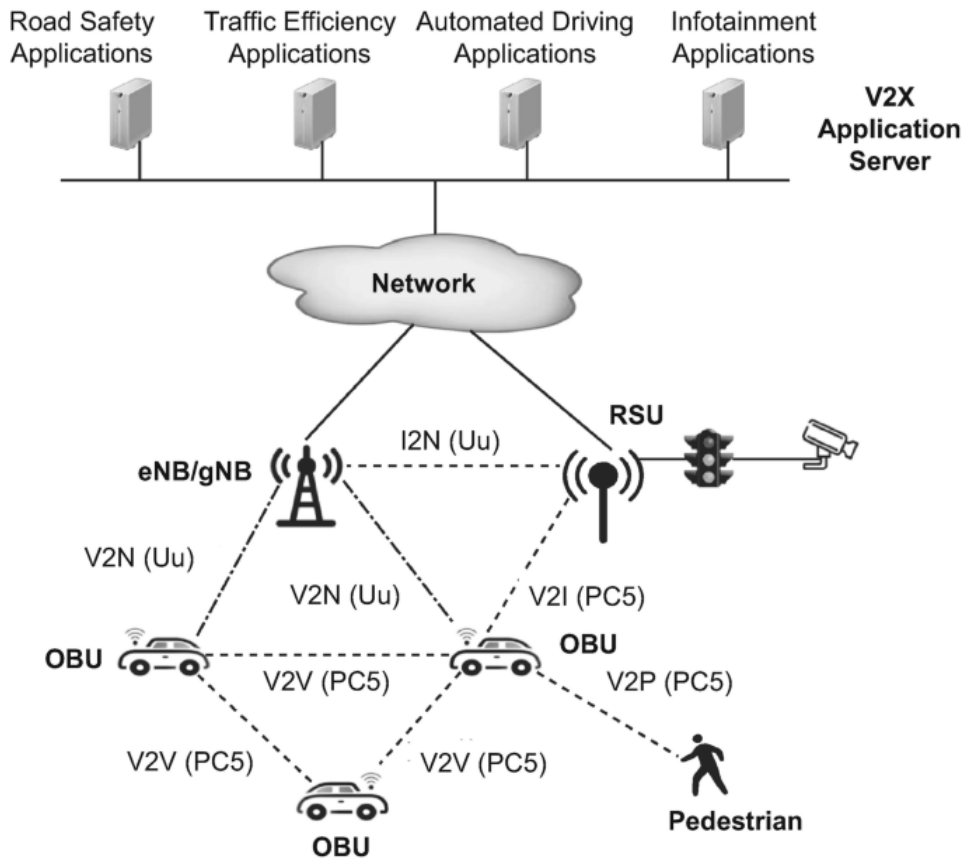


Figure 2.3 LTE-V2X architecture [12]

The specifications of the 3GPP release 14 define two communications modes specifically designed for vehicular communications: Mode 3 and Mode 4. Mode 3 in LTE-V2X refers to the operation within cellular network coverage (In-coverage). In this mode, vehicles communicate through the LTE-Uu interface, utilizing the network infrastructure to facilitate various vehicular communications. This mode is particularly useful for applications that require extended connectivity range and can tolerate higher latencies, such as remote vehicle diagnostics or infotainment services. However, the 3GPP specification do not define a resource management algorithm for this. Each operator can develop and implement its solution that can be one of two categories: dynamic scheduling, Semi-persistent Scheduling (SPS). Further details about these methods are available in [12].

Conversely, Mode 4 represents the operation of network outside the cellular network coverage, primarily using the LTE-PC5 interface. This mode is essential for ensuring continuous vehicle communication in areas where cellular coverage is not available. It is particularly critical for safety-related applications, like ADAS and situational awareness, where direct and low-latency communication between nearby ITS-S are necessary.

### 2.2.1.3 5G and 6G Technologies

Cellular technologies have witnessed dramatic growth over the past two decades. These advancements target to obtain high quality of service with high throughput, low latency and increased reliability. Considering these advances, automotive research institutes and industrial communities show their interest towards exploiting this efficient communication technology to satisfy vehicular services requirements in terms of support of multi-hop, trusted, secure and low latency communications with high reliability. In this context, a growing interest is being shown for considering these cellular networks mainly 5G [13] and 6G [14] and compete with other technologies such as IEEE 802.11p based technologies. The aim of 5G is to promote ultra-low latency, massive machine type communication and enhanced mobile broadband with high throughput. The 5G network capability promises a performance jump compared to 4G in terms of data rate ( $> 1$  Gbps per user; 10 times more compared to LTE), extra low latency (1ms vs. 10ms in LTE), better energy efficiency and etc. Meanwhile, 6G allows for further improved reliability and increased security.

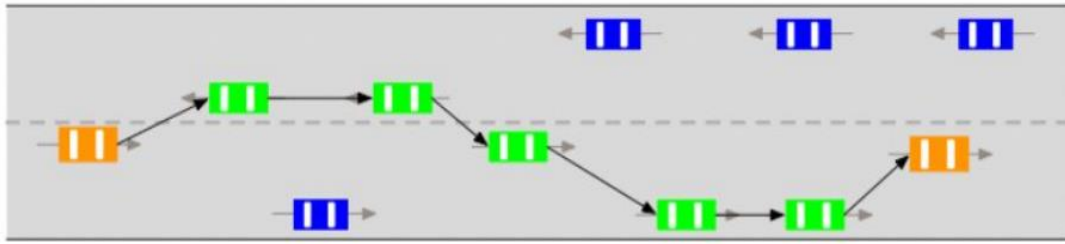
## 2.2.2 Networking and Transport Protocols

This section provides a brief overview of the main networking and transport protocols needed to understand the work proposed in this thesis.

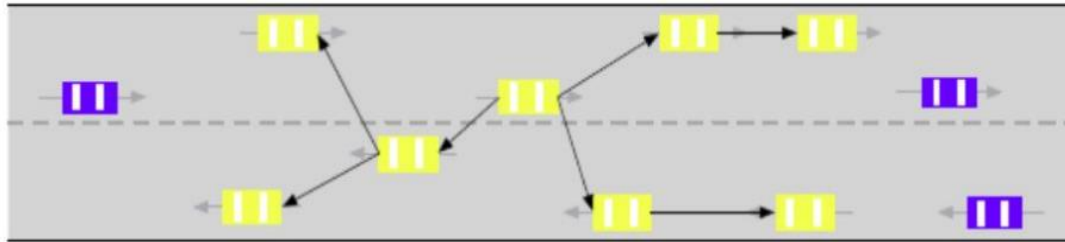
### 2.2.2.1 Networking Protocols

The GeoNetworking Protocol (GN) represents a basic protocol for ITS networking, utilizing geographical coordinates for the targeted dissemination of messages and routing of data packets [15]. It supports a variety of forwarding schemes, demonstrating its versatility in catering to diverse Inter-Vehicle Communication (IVC) application needs, as depicted in Figure 2.4. Briefly, Geo-Unicast is a scheme where messages are directed to a specific destination based on geographic location, ensuring targeted and efficient communication. Geo-Broadcast refers to the broadcasting of messages to all nodes within a defined geographic area, facilitating widespread information dissemination. Lastly, Topologically Scoped Broadcast is a method where messages are broadcast within a topological limit, such as a specified number of hops, allowing for controlled yet extensive message spread within the network.

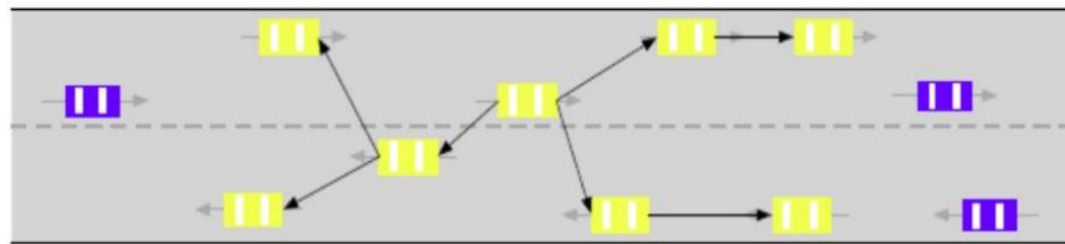
Building upon these functionalities, the integration of GN with Internet Protocol versions 4 and 6 (IPv4/IPv6) adds a layer of scalability and connectivity, ensuring seamless communication across the ITS network [17]. While IPv4 establishes a basic framework for internet communication, IPv6, with its vast address space and enhanced mobility support, is particularly adept at meeting the challenges of dynamic ITS environments.



(a) Geo-Unicast



(b) Geo-Broadcast



(c) Topologically Scoped Broadcast

Figure 2.4 Geo-Networking routing schemes [16]

### 2.2.2.2 Transport Protocols

In the realm of transport protocols within ITS, the Basic Transport Protocol (BTP) stands out as a pivotal protocol [18]. Functionally similar to the User Datagram Protocol (UDP), BTP offers a connectionless, end-to-end, and unreliable transport service. This design is particularly suited for ITS environments, where rapid and flexible communication is more crucial than guaranteed delivery. BTP's primary role involves the multiplexing and demultiplexing of messages at the ITS facilities layer, employing port identifiers in this process. The BTP is distinguished by its two distinct header types, which cater to different communication needs within ITS. The BTP-A header type is designed for interactive packet transport, incorporating both source and destination ports in its structure. This feature is essential for scenarios where bidirectional communication is necessary, such as in V2V and V2I interactions. On the other hand, the BTP-B header type is tailored for non-interactive transport, indicating only the destination port [18]. This streamlined format is beneficial for scenarios like broadcast



communication, where the focus is on disseminating information without necessitating a response.

Moreover, traditional protocols such as UDP and the Transmission Control Protocol (TCP) play integral roles in ITS. While UDP offers a lightweight and fast method for data transmission, ideal for time-sensitive ITS applications, TCP provides a reliable communication channel, ensuring data integrity and order, which is crucial for applications that require accurate and complete data transmission. The use of these protocols in ITS reflects a balanced approach, accommodating a wide spectrum of application and service requirements. For instance, UDP is often preferred for real-time traffic updates, while TCP is utilized for scenarios where the accurate and complete transfer of information is paramount, such as in the exchange of detailed map data.

## **2.2.3 Management and Security Entities**

In the ITS-S architecture, the Management and Security cross-layers play critical roles that transcend traditional layer boundaries. These cross-layers are not confined to a single aspect of the network but interact with and influence multiple layers to ensure comprehensive system management and robust security.

### **2.2.3.1 Management Cross-Layer**

This layer encompasses a range of functionalities that ensure efficient network operation. Central to this is the Management Information Base (MIB), a repository responsible for storing global communication stack parameters [19]. This database is crucial for maintaining and accessing vital network management data. The management layer's operation is further guided by several interface standards [20], [21], and [22], which dictate how data is accessed and used across different network layers, ensuring standardized and coherent management practices. Additionally, this layer plays a crucial role in congestion control, actively managing network traffic to prevent bottlenecks and maintain optimal performance. Another key function is facilitating inter-layer communication, enabling different layers of the ITS architecture to interact and coordinate, which is essential for the network's adaptive and dynamic response to varying traffic conditions and demands. The management layer also oversees regulatory compliance, station operation, and application functionality within ITS, ensuring that all aspects of the system adhere to legal standards, remain operational, and meet both user needs and system requirements.

### **2.2.3.2 Security Cross-Layer**

The security cross-layer in ITS is essential for protecting the network against a wide range of cyber threats and ensuring data integrity and privacy. A notable feature of this layer is the intrusion management service, accessible to all network layers [23], which monitors for and addresses security breaches, providing a comprehensive defense mechanism against potential intrusions. Additionally, this cross-layer implements strategies for pseudonym management and

ITS-S authentication using certificates [2], ensuring secure and verified communication within the network. This is particularly important for maintaining trust and confidentiality in ITS communications. It also focuses on hardware security, safeguarding physical components like sensors and communication devices from tampering and unauthorized access through measures such as secure boot processes and hardware-based encryption. Furthermore, extensive surveys on the privacy and security aspects of IVC are available in [25], [26], and [27]. These studies provide in-depth analysis and insights into the evolving challenges and solutions in the field of ITS security, highlighting the ongoing need for robust and adaptable security measures in this rapidly advancing area.

## **2.2.4 C-ITS Services and Related Messages**

In the context of C-ITS, services and messages are foundational elements that facilitate the seamless communication and interaction between ITS-S within the transportation network. This section introduces the primary services and corresponding messages that are integral to the framework of this thesis.

### **2.2.4.1 Cooperative Awareness Service**

The Cooperative Awareness (CA) basic service is a fundamental aspect of C-ITS, focusing on augmenting road safety and enhancing traffic efficiency through the generation, processing, and management of CAMs, as detailed in references [28], [29], and [30]. This is achieved by periodically broadcasting CAMs to all ITS-Ss within communication range via V2X communications. CAMs contain essential status information such as speed, position, time, and various vehicle attributes, including type and dimensions. This data enables applications like collision avoidance systems to assess the proximity and movement of nearby vehicles, thus aiding in the prevention of accidents.

The generation of CAMs adheres to a strict frequency regime, ranging between a minimum of 100 ms and a maximum of 1000 ms. The criteria for triggering the generation of a CAM are based on specific changes in the vehicle's dynamics: a positional change exceeding 4 meters, a speed variation greater than 0.5 m/s, a heading alteration over 4 degrees, or a one-second interval since the last CAM transmission. The generation of CAMs also depends on the channel congestion status. For example, in high-density scenarios, the CA service adjusts the CAM generation frequency to align with the remaining capacity of the radio channel, thereby mitigating channel congestion. In the case of ITS-G5, the CA service manages CAM generation in accordance with the DCC guidelines as specified in ETSI TS 102 724 [31], while in C-V2X systems, DCC is managed differently, as outlined in ETSI TS 103 574 [32].

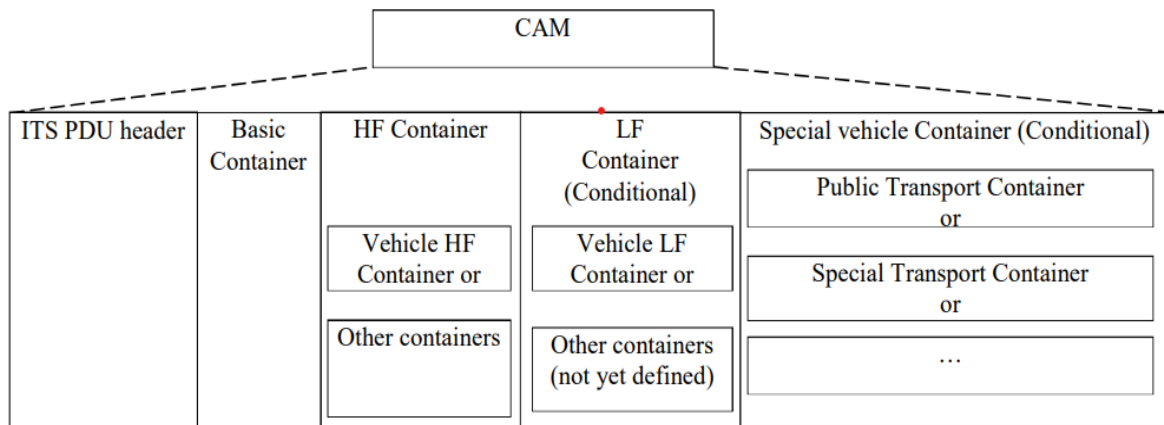


Figure 2.5 General structure of a CAM [30]

As illustrated in Figure 2.5, the structure of the CAM is defined using the. It encompasses several key components: an ITS PDU header, which specifies the protocol version and ITS-S ID; a Basic Container, detailing the station type and its geographic position at the time of CAM generation; a High-Frequency (HF) Container, carrying dynamic or fast-changing vehicle information; a Low-Frequency (LF) Container, providing static or slow-changing vehicle data; and a special vehicle container, which includes supplementary information tailored to the vehicle’s specific role within the ITS network. This format ensures standardization and effective communication across various ITSs, enhancing the overall functionality and safety of within the C-ITS framework.

#### 2.2.4.2 Decentralized Environmental Notification Service

The central role of the Decentralized Environmental Notification (DEN) service [33] involves the generation, management, and processing of the DEN Message (DENM), which is initiated by an ITS-S application in response to road hazards or abnormal traffic conditions. Unlike the CAM, which is periodic, the DENM is an event-triggered message designed to warn users about hazardous events, such as road accidents or the approach of emergency vehicles. Initiated by an ITS-S application in response to specific road hazards or abnormal traffic conditions, each DENM contains critical information about the incident’s nature and location. Once created, the DEN Service ensures the DENM’s delivery to the ITS networking and transport layer, facilitating its broad dissemination, typically via multi-hop communications, to all ITS-Ss within the affected geographic area. At the receiving end, DEN processes these DENMs and relays their content to the relevant ITS-S application, which then evaluates the information’s relevance and, if pertinent, presents it to the driver. This process aids in informed decision-making, allowing users to appropriately respond to situations, such as yielding to an approaching ambulance.

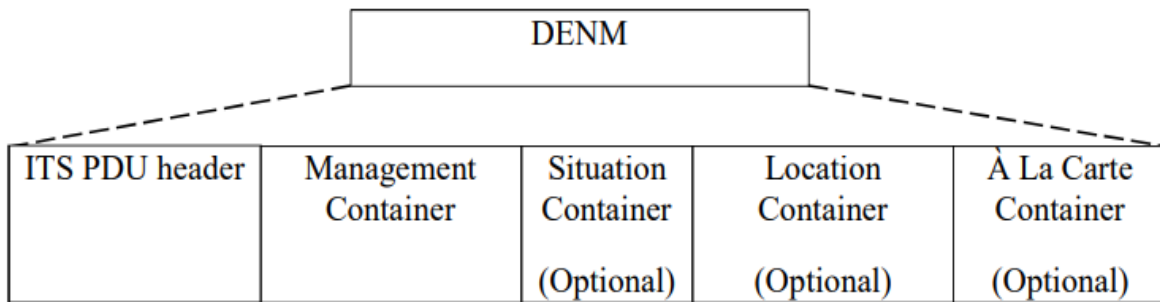


Figure 2.6 General structure of a DENM [33]

The structure of the DENM, outlined in Figure 2.6, consists of several key components: an ITS PDU header, which specifies the protocol version, message type, and station ID; a Management container, encompassing the DENM’s management information; a Situation container, describing the detected event; a Location container, pinpointing the event's location; and an a la carte container, providing additional information not covered in the other containers. The inclusion of these containers depends on the type of ITS-S transmitting the DENM. More comprehensive details about these containers can be found in [33]. This structured approach in the DEN Service enhances situational awareness among road users and supports safe and efficient navigation in response to various road events.

### 2.2.4.3 Vulnerable Road User Service

The Vulnerable Road User (VRU) service in C-ITS plays a vital role in ensuring the safety of road users who are more susceptible to traffic hazards, such as pedestrians, cyclists, motorcyclists, and animals [34]. This service, integral to the facilities layer of the ITS-S architecture. This service is a key component of the facilities layer within the ITS-S architecture and is responsible for the development and management of the VRU Awareness Message (VAM) [34].

Through the dissemination of VAMs, the VRU service aims to increase the situational awareness of all road users, particularly drivers of motorized vehicles, thereby reducing the likelihood of accidents and improving overall road safety. Each VAM contains critical information regarding the VRU, including details such as their current position, motion state, and specific attributes like type and dimensions, as depicted in Figure 2.7. The content of these messages is adapted according to the VRU profile, categorizing individuals into distinct groups such as pedestrians, bicyclists, or motorcyclists. Each group has unique characteristics and safety requirements as defined in [35]. The process of generating and disseminating VAMs starts with the originating ITS-S composing the message, followed by its distribution to other ITS-Ss within its effective communication range. This ensures that vehicles in the vicinity are promptly informed about the VRUs’ presence, empowering drivers to make informed and timely decisions to prevent potential collisions. Additional detailed specifications and use cases regarding the VRU service can be found in [36].

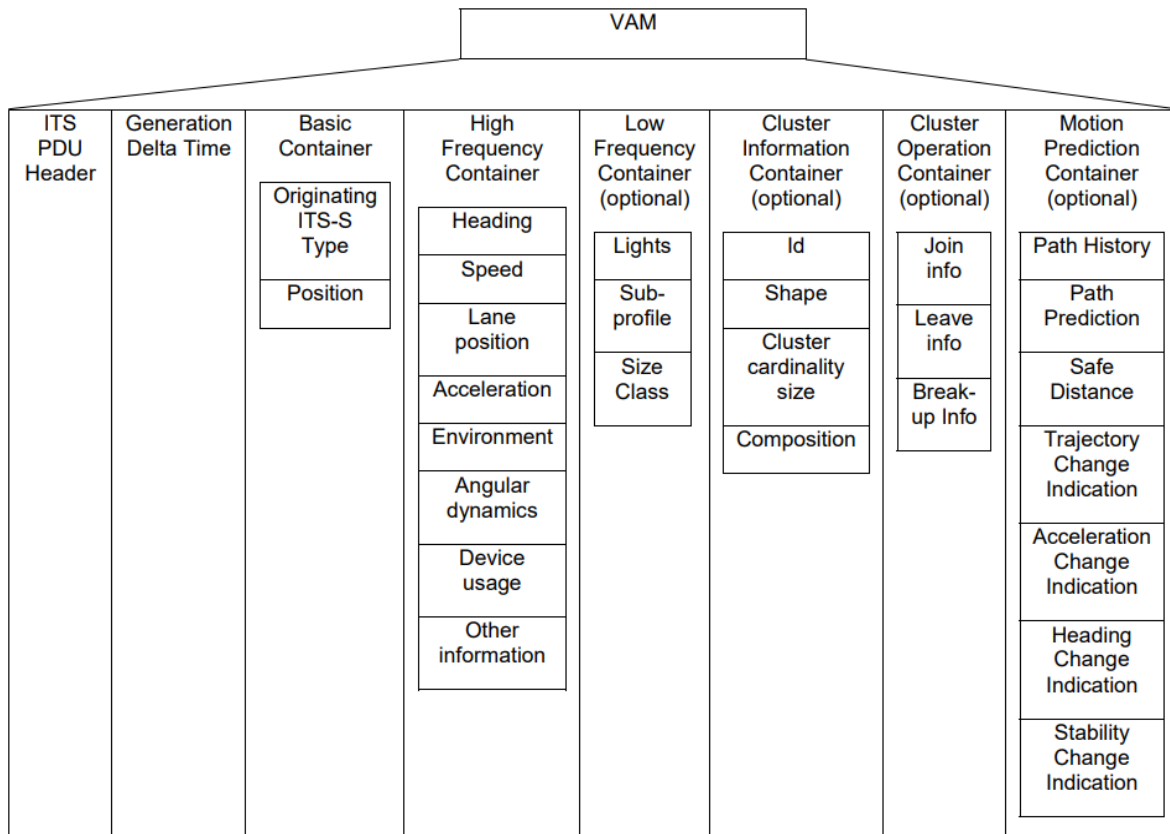


Figure 2.7 General structure of a VAM [34]

#### 2.2.4.4 Cooperative Perception Service

While services like CA and DEN play crucial roles in enhancing road safety by providing vital information about vehicles and road conditions, they have inherent limitations in terms of perception scope and detail about the driving environment. These services typically rely on data directly obtained from individual vehicles or infrastructure, which may not always provide a comprehensive view of the surroundings, especially in complex urban environments or in situations with obstructed views.

The CP service addresses these limitations by enabling a more holistic perception of the environment [37]. It involves the sharing of sensory data among vehicles and infrastructure, allowing for a collective and enhanced understanding of the road situation. This service goes beyond the capabilities of individual vehicle sensors, combining data from multiple sources to create a richer, more accurate, and more reliable perception of the environment. This comprehensive perception is crucial for advanced applications, such as supporting autonomous driving systems in making safer navigation decisions and providing drivers with enhanced situational awareness.

As the core of this thesis revolves around the significance of CP for CAVs within the realm of C-ITS, the next chapter is dedicated to provide an in-depth exploration of its structural

framework and functionalities. This study aims not only to introduce CP's operational mechanisms and pivotal role in addressing the limitations inherent in the existing C-ITS services but also to examine its potential limitations, creating a pathway for introducing the contributions proposed in this thesis.

#### 2.2.4.5 Other Related Services

In addition to the core services in C-ITS, several other related services play a pivotal role in enhancing the functionality and safety of road transportation [38]. These services, including but not limited to the Local Dynamic Map (LDM) service and Signal Phase and Timing (SPAT) service, contribute significantly to the comprehensive management of traffic and road infrastructure. Briefly, the LDM Service acts as a dynamic repository, aggregating data from various sources to create a detailed map of the road environment, integrating information from messages like CAMs, CPMs and DENMs. This comprehensive map supports ITS applications by providing real-time traffic conditions. The SPAT Service focuses on disseminating traffic signal timing and phasing information at intersections through SPAT messages, crucial for adaptive traffic control.

### 2.2.5 Decentralized Congestion Control

ETSI defines the DCC as an essential component for ITS-S within the ITS-G5 network to mitigate network congestion [39]. DCC operates as a cross-layer function, encompassing components across various layers of the C-ITS architecture, as depicted in Figure 2.8:

- **DCC\_ACC:** Located in the access layer, it operates as a gatekeeper to control the traffic that is effectively transmitted by each ITS-S.
- **DCC\_NET:** Located in the networking and transport layer, it enables ITS-S to exchange information about the channel load, enable each ITS-S to be aware of the channel load experienced by its one-hop and two-hop neighbors.
- **DCC\_FAC:** Located in the facility layer, it controls the number of messages generated by each application/service within each ITS-S. The control considers the messages' traffic classes or DCC profiles (DPs). Thus, DCC\_FAC distributes access to the channel among the different applications/services within each ITS-S.
- **DCC\_CROSS:** Located in the management layer, defines the necessary management support functions for DCC and the required interface parameters between the DCC management entity and the DCC entities in the Facilities, the Networking & Transport and the Access layers.

A specific focus will be taken on the DCC\_ACC and DCC\_FAC, since they contain the main mechanisms that control congestion and that can affect the communication performance.

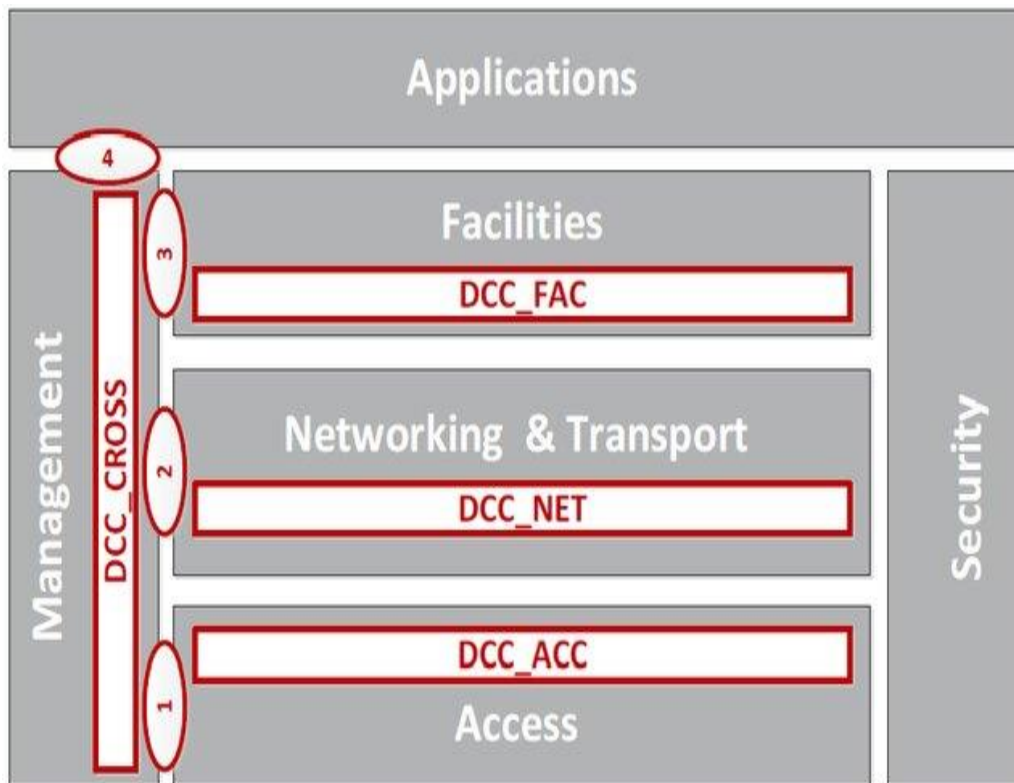


Figure 2.8 DCC architecture [39]

### 2.2.5.1 DCC Access

The DCC\_ACC adapts the amount of time that each ITS-S can access the channel as a function of the channel load. The channel load used as input for the algorithm can be the one locally measured by an ITS-S or the one provided by DCC\_NET if the ITS-Ss share their channel load measurements. The specifications provide in [40] defines the DCC\_ACC component 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 traffic class. The traffic class is defined by the Facilities layer to provide different priority levels to different types of messages. Four different traffic classes are differentiated by DCC Access and mapped to four DPs: DP0, DP1, DP2 and DP3, where DP0 has the highest priority. At the lower layers, these DCC profiles are mapped to the EDCA access categories of ITS-G5.
- **Queuing:** DCC Access implements 4 different queues, one for each traffic class or DCC profile. 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: 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.

Basically, two approaches are defined by the DCC Access to control the rate of packets transmitted per vehicle: Reactive and Adaptive [40]. Both algorithms adapt the time between consecutive packet transmissions based on the channel load or CBR. The CBR is defined as the percentage of time that the channel is sensed as busy.

#### DCC Reactive Approach

The Reactive approach makes use of a state machine for flow control, as depicted in Figure 2.9. Each state is mapped to a range of CBR values and to a time, as illustrated Table Table 2.3.

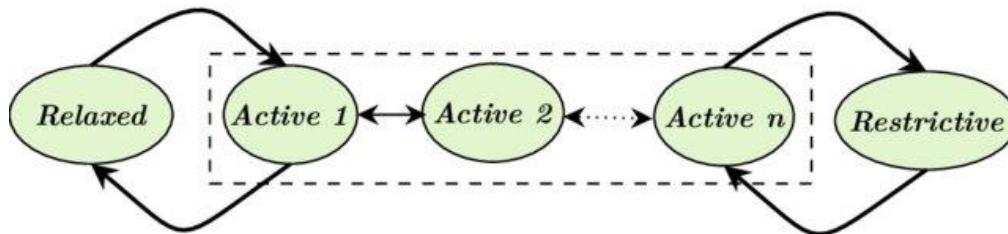


Figure 2.9 Reactive DCC state machine [41]

The DCC Reactive approach uses three primary states: Relaxed, Active, and Restrictive. The changing between these states is based on a predefined CBR values, as indicated in Table 2.3. During the Relaxed state, the network experiences relatively low congestion levels, and ample bandwidth is available for data transmission. In this phase, the DCC approach adopts a more permissive and flexible stance toward data transmission. This flexibility may entail larger contention windows, shorter inter-frame spaces, or higher data transmission rates, tailored to the specific network configuration.

Conversely, the Restrictive state entails stringent measures within DCC to manage and alleviate congestion. Such measures may involve reducing the contention window size, increasing inter-frame spaces, or lowering data transmission rates to maintain network stability and prevent degradation.

#### DCC Adaptive Approach

The Adaptive approach of the DCC uses a linear control process for flow control. The process is designed so that each ITS-S adapts its packet transmission rate in order for the channel load to converge to a target value  $CBR_{target} = 68\%$ .



States	CBR (%)	Frequency (Hz)	Toff (ms)
Relaxed	< 30	10	60
Active 1	30 to 39	5	100
Active 2	40 to 49	2.5	200
Active 3	50 to 60	2	250
Restricted	> 60	1	1000

Table 2.3 Reactive DCC parameter look-up table [41]

### 2.2.5.2 DCC Facilities

DCC Access controls the total amount of messages that an ITS-S can transmit per second. DCC at the Facilities layer controls the number of messages that each application/service can generate [42] to satisfy the DCC Access limit imposed to each ITS-S. To this aim, the DCC\_FAC makes use of the DCC Access limit, the message size and the message interval from each application/service.

## 2.2.6 C-ITS Applications

C-ITS applications are intricately designed to enhance road safety, environmental sustainability, and traffic efficiency. By augmenting situational awareness among drivers about imminent road hazards through real-time alerts, these applications significantly contribute to the prevention of collisions and the mitigation of unforeseen dangers. Additionally, C-ITS plays a vital role in reducing environmental impact and alleviating congestion by optimizing fuel usage and decreasing travel times. Broadly, C-ITS applications are categorized into three primary domains: Traffic Safety, Traffic Efficiency, and Infotainment.

### 2.2.6.1 Traffic Safety Applications

In the domain of C-ITS, Traffic Safety Applications play an instrumental role in safeguarding lives on the road. These applications are pivotal in enhancing road safety by significantly reducing vehicular accidents. Employing advanced technologies like cooperative forward collision warnings and pre-crash alerts, they provide drivers with real-time, comprehensive situational awareness. For instance, Cooperative Forward Collision Warnings utilize real-time data on vehicle positioning, speed, and direction to predict and alert drivers about potential collision scenarios, facilitating quick corrective actions. In a similar vein, Pre-Crash Alerts and Hazardous Location Notifications focus on delivering timely warnings about

imminent dangers such as roadblocks or construction zones, enabling drivers to proactively respond to avoid hazardous situations. The life-saving potential of these applications is underpinned by their ability to function with low latency and high reliability, ensuring that alerts are both timely and accurate.

### 2.2.6.2 Traffic Efficiency Applications

This type of applications is designed to optimize traffic flow and enhance overall road management. These applications leverage a wealth of data, including both real-time and historical road conditions, to improve travel efficiency and reduce environmental impact.

### 2.2.6.3 Infotainment Applications

Their aim is to elevate the driving experience by providing a variety of entertaining and informative services. These applications transform the vehicle into a connected hub, where drivers and passengers can engage in activities like multimedia downloads and peer-to-peer exchanges. This functionality allows for the sharing and downloading of multimedia content, such as music and videos, either with other vehicles or infrastructure nodes, enriching the in-vehicle entertainment options.

## 2.3 Paradigms and Innovation in C-ITS

This section presents the key emerging paradigms and innovation technologies within the C-ITS framework.

### 2.3.1 Emerging Paradigms

The key emerging paradigms within the C-ITS framework principally lies in computing paradigms.

#### 2.3.1.1 Cloud and Fog Computing

In C-ITS, cloud computing provides a powerful, scalable platform for extensive data storage, networking, and software services accessible via the internet. It supports a wide range of C-ITS applications, from traffic management tools to smart urban infrastructure solutions, by enabling efficient long-term data analysis and resource-intensive computational tasks [43]. The flexibility and scalability of cloud computing are instrumental in the rapid development and deployment of various C-ITS applications, ensuring that they can adapt to evolving traffic patterns and urban needs.

Complementing cloud computing, Fog Computing enhances C-ITS by bringing computational capabilities closer to the edge of the network [44]. This approach is particularly effective in managing the 'big data' generated by C-ITS, addressing key challenges such as bandwidth limitations and latency, which are crucial for the high performance of C-ITS

applications. By enabling real-time data processing and aggregation during transit, Fog Computing addresses the latency issues associated with processing large volumes of data in distant cloud data centers. This is essential for C-ITS applications that require immediate data analysis and response, such as dynamic traffic signal control and V2X communication for autonomous vehicles.

### 2.3.1.2 Edge Computing

Edge Computing (EC) in C-ITS decentralizes computational resources to the network's user end [45]. By minimizing the distance between data sources and processing nodes, EC significantly reduces latency, which is vital for applications requiring immediate response, such as autonomous vehicles and smart traffic control systems.

## 2.3.2 Technological Innovations in C-ITS

The progression of C-ITS is based on a series of groundbreaking technological innovations, including agent/multi-agent-based systems, machine learning, and reinforcement learning. These innovations lay the foundation for more efficient, responsive, and intelligent transportation networks.

### 2.3.2.1 Agent/Multi-Agent-Based Systems

Agent-based systems encompass the deployment of autonomous agents as software entities capable of performing independent actions in a networked environment. These agents, designed to respond to environmental changes or inputs from other agents, offer a versatile framework for modeling complex systems comprising multiple interacting components.

In the context of C-ITS, autonomous agents can represent a wide range of entities, including vehicles, traffic lights, sensors, and traffic controllers. Each agent operates autonomously, evaluating real-time situations and making decisions that contribute to the overall efficiency and safety of the traffic management system. Furthermore, when multiple agents interact within a networked environment, they collectively form a multi-agent system. This is particularly significant in C-ITS, where numerous vehicles, pedestrians, and infrastructure components communicate and coordinate with each other.

Additionally, multi-agent systems play a critical role in optimizing traffic flow, reducing congestion, and enhancing road safety. They find applications in dynamic route planning, adaptive traffic signal control, and the management of V2X communications, where each agent may represent a node in the vehicular network. These systems often employ various learning paradigms to enable agents to adapt and make informed decisions in complex traffic scenarios.

### 2.3.2.2 Machine Learning

ML provides a powerful set of tools for analyzing extensive datasets and extracting valuable insights essential for informed decision-making. ML algorithms are effective in mining vast

amounts of traffic data from various sources, including sensors and V2X communications. They identify intricate patterns, discern trends, and offer predictive capabilities that are indispensable for tasks such as predictive traffic modeling, accident prevention, and congestion analysis.

Moreover, ML can be instrumental in predictive maintenance, allowing for the anticipation of potential issues in both vehicles and infrastructure components. This allows for reduced downtime, enhanced system reliability, and improved overall operational efficiency.

### 2.3.2.3 Reinforcement Learning

RL is a key player in C-ITS, enabling agents to acquire optimal decision-making strategies through interactions with their environment [46]. RL's primary application in C-ITS revolves around the development of adaptive systems, extending its influence across different layers of the ITS-S reference architecture. In this context, RL algorithms enable agent-based systems to dynamically adapt to real-time data inputs from various sources, responding effectively to ever-changing environmental conditions. Such adaptive systems play a vital role in optimizing traffic flow, mitigating congestion, and enhancing travel times. Additionally, RL empowers C-ITS systems to make dynamic decisions, adjusting to changing traffic conditions and providing real-time responses to incidents, roadwork, and varying traffic demands. This adaptability and responsiveness to real-world traffic scenarios solidify RL's role in addressing complex transportation challenges and ensuring the effectiveness and resiliency of C-ITS services.

## 2.4 Conclusion and Future Directions of C-ITS

In conclusion, it's evident that the field is rapidly evolving, driven by a blend of innovative technologies and emerging paradigms. This chapter provided an in-depth understanding of the present status and the potential future developments of C-ITS. It introduced its fundamental components and architectures while placing emphasis on significant areas of interest such as cloud and edge computing, machine learning, and emerging communication technologies and C-ITS services.

The exploration of various technological innovations within the C-ITS framework points towards a future where transportation systems are not only more interconnected and intelligent, but also increasingly tailored to individual users' needs. Advancements in autonomous vehicle technology, enhanced connectivity, augmented road perception, and the development of powerful paradigms for AI-based analytics are set to significantly improve the efficiency, safety, and sustainability of transportation systems.

The next chapter will explore the concept of CAP within C-ITS, offering a comprehensive examination of various sensing technologies and service specifications, with a particular focus on their application to CAVs and roadside infrastructure.

# Chapter 3

## Cooperative and Augmented Perception in C-ITS: An Overview

### 3.1 Introduction

This chapter focuses on three different parts. First, it offers an in-depth exploration of AVs, detailing their levels of automation, sophisticated sensing technologies, and inherent limitations. Second, it introduces the concept of CAP, highlighting its benefits in enhancing road safety and efficiency. Third, the chapter highlights the essence of resilience for this concept, exploring the various challenges in areas such as data quality, security, and communication reliability.

### 3.2 Autonomous Vehicles and Perception

The first part of this chapter introduces the concept of AVs and road perception within C-ITS.

#### 3.2.1 Overview on Autonomous Vehicles

AVs, or self-driving vehicles, represent a significant leap in automotive technology. These vehicles integrate complex systems to navigate and make decisions in various environments safely. This section will provide a comprehensive overview of an autonomous vehicle, covering its basic components, the levels of automation, and the workflow of its system architecture

##### 3.2.1.1 Definition

An AV, often referred to as self-driving or driverless car, is a form of transportation equipped with advanced technology that enables them to operate and make decisions without direct human intervention. These vehicles leverage a combination of sensors, algorithms, AI, and control systems to perceive their environment, plan routes, and execute maneuvers, as depicted in Figure 3.1, all with the primary objective of ensuring safe and efficient transportation.



Figure 3.1 Example of an AV

### 3.2.1.2 Automation Levels

AVs are classified into different levels of automation, ranging from Level 0 (no automation) to Level 5 (full automation) [47], as illustrated in Figure 3.2:

- Level 1 - Driver Assistance: The vehicle helps either in longitudinal or lateral control, but not both simultaneously. The driver is required to continuously monitor the system and be ready to take over full control at any time. An example is Adaptive Cruise Control (ACC), where the vehicle controls the speed and maintains a safe distance from the vehicle ahead.
- Level 2 - Partial Automation: The vehicle can take over both longitudinal and lateral controls under certain conditions for a limited time. The driver must still monitor the system continuously and be prepared to take control when necessary. Systems like Highway or Traffic Jam Assistants exemplify this level, managing speed and direction in specific scenarios like highways or traffic jams.
- Level 3 - Conditional Automation: The vehicle manages both steering and speed in certain situations without requiring the driver's constant attention. However, the driver must still be ready to take control when the system requests or in case of a failure. A typical use case is the Highway Pilot, allowing the vehicle to handle driving tasks at specified speeds.
- Level 4 - High Automation: The vehicle performs all driving tasks in specific conditions. If the driver doesn't respond to a takeover request, the vehicle can safely manage the situation, like bringing itself to a stop. An example is Remote Valet Parking, where the vehicle autonomously navigates to a parking spot without a driver present.
- Level 5 - Full Automation: The highest level of automation, where the vehicle is capable of performing all driving functions under all conditions. Passengers do not need to

intervene or monitor the system at any point. Automated Taxis represent this level, where vehicles operate without any driver input throughout the journey.






















For on-road vehicles		 Human driver	 Automated system		
		Steering and acceleration/ deceleration	Monitoring of driving environment	Fallback when automation falls	Automated system is in control
Human driver monitors the road	<b>0</b> NO AUTOMATION				N/A
	<b>1</b> DRIVER ASSISTANCE				SOME DRIVING MODES
	<b>2</b> PARTIAL AUTOMATION				SOME DRIVING MODES
Automated driving system monitors the road	<b>3</b> CONDITIONAL AUTOMATION				SOME DRIVING MODES
	<b>4</b> HIGH AUTOMATION				SOME DRIVING MODES
	<b>5</b> FULL AUTOMATION				

Figure 3.2 Standard automation levels [47]

### 3.2.2 Recent Advances in Sensing Technologies

Perception sensors, mounted on different positions of the vehicle, are used to sense the environment and provide a representation of the vehicle’s immediate driving context. An example of sensors is depicted in Figure 3.3.

#### 3.2.2.1 Radar

Radar (Radio Detection and Ranging) uses radio waves to detect and locate objects in its vicinity. It operates on the principle of emitting radio waves (often in the form of radio frequency pulses) and then measuring the time it takes for these waves to bounce off objects and return to the sensor. By analyzing the reflected signals, Radar can determine the distance, speed and direction of objects within the FoV of the vehicle.

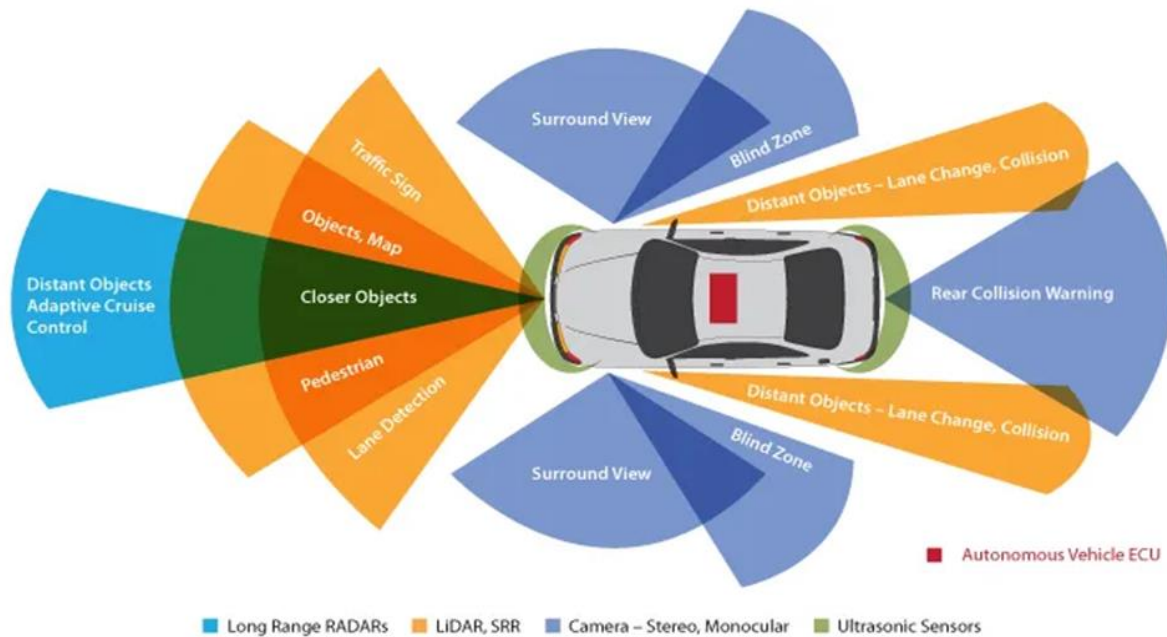


Figure 3.3 Sensors of an autonomous vehicles [48]

### 3.2.2.2 Lidar

Lidar (Light Detection and Ranging) operates by emitting laser pulses and measuring the time it takes for these pulses to bounce back after striking objects. This process enables it to perform accurate distance measurements and create detailed and varied dimensional representations of its surroundings, over distances up to 200 m [49]. Lidar technologies come in various types, such as 2D and 3D, each suited for specific applications [49]. In 2D Lidar, a single laser beam is spread out using a rotating mirror to create a two-dimensional view. Meanwhile, 3D Lidar systems, which can include between 4 and 128 lasers, offer a more complex representation by achieving a 360-degree horizontal and 20 to 45-degree vertical FoV, providing remarkable accuracy within a few centimeters [50] and [51].

### 3.2.2.3 Camera

Camera sensors play a fundamental role in the recognition and understanding the driving environment. This type of sensors is broadly categorized into two main types: visible-light-based and infrared-based cameras, depending on the wavelength of their operation. The operational range of camera sensors typically spans up to approximately 250 m, although the extent of their effectiveness is contingent upon the quality of the lens employed. Visible-light cameras, functioning within the spectrum of human vision between 400 and 780 nanometers, which are further segmented into three color bands: Red, Green, and Blue (RGB). To achieve depth perception and facilitate stereoscopic vision, AVs often integrate pairs of visible-light cameras with known focal lengths.



While camera sensors offer the advantage of accurately capturing and recording environmental attributes such as texture, color distribution, and object contours, their FoV is limited due to the narrow angle of observation dictated by the camera lens. Consequently, the integration of multiple cameras has become crucial in AVs to provide comprehensive and panoramic surveillance of the surroundings [52] and [53].

#### 3.2.2.4 GNSS/GPS

The Global Navigation Satellite System (GNSS) and Global Positioning System (GPS) are used by AVs to determine their precise location. However, these technologies can introduce positional errors with a mean value of approximately 3 m and a standard deviation of 1 m [54]. This margin of error can further increase in complex urban environments, reaching up to 20 m [55] and [56].

#### 3.2.2.5 Performance Comparison Between Sensors

The selection of appropriate sensors is crucial for optimizing the performance of AVs across a variety of driving conditions. Each sensor presented above brings unique strengths and faces distinct challenges, significantly influencing their utility for accurate perception. GNSS/GPS systems are notable for their autonomy, capable of functioning without a pre-established starting point; however, their effectiveness is limited by reliance on satellite visibility. These systems are vulnerable to the urban canyon phenomenon, where high-rise buildings obstruct satellite signals, leading to significant positional inaccuracies. Radar technology stands out for its proficiency in detecting objects over long distances and accurately measuring their speed. Still, its utility is diminished by a lack of fine angular resolution, which is crucial for detailed object identification and classification. Lidar sensors strike a balance, offering precise measurements of physical distances through laser beams, however, they fall short in capturing object textures, a limitation attributed to the sparse nature of the generated point cloud data. Cameras are known by their ability to discern color and texture, bolstering segmentation and object classification efforts. Nonetheless, they are limited in measuring depth and are susceptible to impairments caused by adverse weather or fluctuating lighting conditions.

These sensor-specific limitations underscore the necessity for advanced solutions, such as multi-sensor data fusion techniques.

### 3.2.3 Multisensory Local Data Fusion

The integration of multi-sensor local data fusion in AVs is a crucial strategy for enhancing their perception systems. This approach directly tackles the limitations inherent in individual sensors, bolstering both data accuracy and reliability. By fusing data from varied sensors, AVs can attain a more comprehensive understanding of their surroundings, resulting in improved decision-making and safer navigation. Furthermore, the fusion process significantly enhances accuracy in object detection, classification, and environmental modeling, reducing the risk of

incorrect interpretations and decisions [57]. The fusion of data from diverse can take place at various levels [58]. The following sections provide an overview of each level.

### 3.2.3.1 Low Level Fusion

At the low data level, multisensory fusion involves the direct improvement of unprocessed sensor data. This level focuses on combining the raw inputs from various sensors without any preliminary processing. The main challenge at this level is managing and interpreting the massive volume of diverse sensor data. Fusion at this level can provide a detailed and rich dataset, which is essential for precise localization [59], object detection [62], and environmental mapping. However, it requires substantial computational resources to effectively merge and utilize the raw data for real-time applications

### 3.2.3.2 Feature Level Fusion

Feature-level fusion takes a step beyond raw data fusion by extracting relevant features from each sensor's data. These features may include object positions, velocities, shapes, and other characteristics that are useful for perception and decision-making tasks. By combining these features from multiple sensors, AVs can obtain a more refined and informative representation of their surroundings [61]. Feature-level fusion enhances object detection and classification accuracy, enabling to differentiate between various objects, such as pedestrians, vehicles, and road signs, with greater precision [62].

### 3.2.3.3 High Level Fusion

At the object level, the fusion process involves integrating the outcomes of individual sensor analyses after the classification task. This means that each sensor independently detects and classifies objects, and these detections are then fused to provide a unified and accurate representation of the environment. This approach is beneficial in scenarios where different sensors have varying capabilities in detecting certain types of objects. For instance, radar is effective in detecting vehicles at a distance, while cameras are better at classifying pedestrian features. By fusing these independent detections, AVs can achieve a more reliable and comprehensive understanding of their surroundings [63].

## 3.2.4 Limitations of Individual AV Perception System

This section assesses the principal challenges and shortcomings related to the perception system of a single AV.

### 3.2.4.1 Environmental Sensitivity and Blind Spots

One of the major challenges for AVs perception systems is their susceptibility to adverse weather conditions, such as heavy rain, fog, or snow. These conditions can significantly degrade

the performance of onboard sensors like Lidar, Radar, and cameras, affecting their reliability and, consequently, the efficiency of the sensor fusion process [64]. Particularly, sensors, such as Lidar and cameras, often have a limited FoV, leaving certain areas outside their detection range. This limitation is particularly critical in urban environments, where pedestrians or vehicles may unexpectedly emerge from the blind spots [65].

### 3.2.4.2 Data Processing, Cost, and System Complexity Challenges

The effectiveness of AV perception systems necessitates advanced data processing and fusion techniques to ensure timely and accurate interpretation of sensor inputs. Delays or inefficiencies in these processes can lead to dire consequences, underscoring the need for further high-performance computing solutions.

### 3.2.4.3 Positioning Inaccuracies and Vulnerability to Sensor Spoofing

Single perception systems are susceptible to security vulnerabilities, including sensor spoofing and cyberattacks. These malicious actions can manipulate sensor data, leading to incorrect navigational decisions. For instance, an attacker might create phantom obstacles in radar data or alter Lidar data to deceive the AV's perception system.

To that end, the adoption of V2X communication represents a strategic and effective approach to augment the environmental perception of AVs. Within the context of CAVs, the implementation of various services for diverse information exchange significantly mitigates the challenges faced by individual AV systems, however still limited in providing information about the road environment.

## 3.3 Cooperative and Augmented Perception

This section elaborates on the concept of CP and augmented CP within C-ITS, including its foundational principal, benefits, and standardization efforts.

### 3.3.1 Definition and Benefits of CP

#### 3.2.1.1 Definition

At its core, CP within C-ITS involves the exchange of sensor data among CAVs, as defined by Figure 3.4. This exchange is facilitated through varied communication scenarios, including V2V, V2I, using advanced wireless communication technologies such as ITS-G5 and C-V2X [66] and [67]. This process empowers CAVs to collaboratively augment their environmental

awareness, principally overcoming the limitations of their onboard sensors. Specifically, this process receives perception information from different sources, such as onboard sensors and the local database, which is used to store perception V2X communication objects, as depicted in Figure 3.5. The received data from different sources is then encapsulated into standardized messages for sharing with other CAVs.

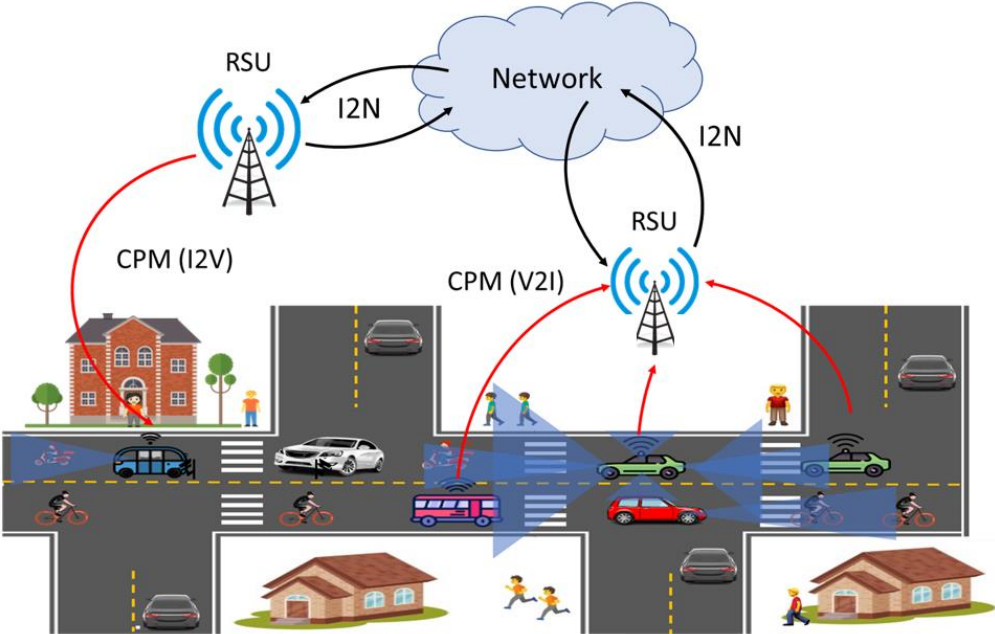


Figure 3.4 The concept of CP in an urban area

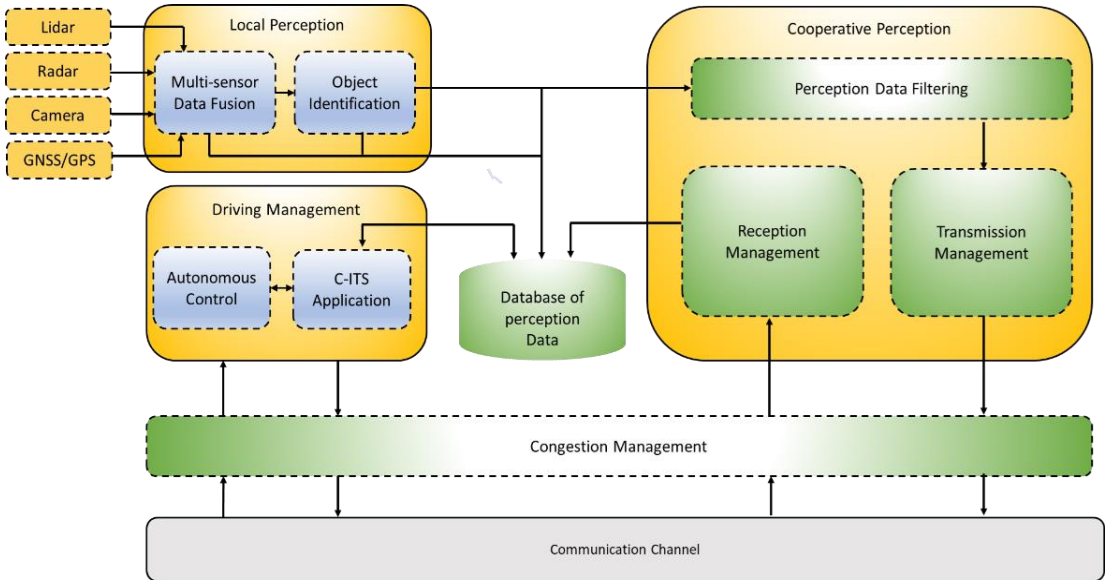


Figure 3.5 Basic architecture of CP

### 3.3.2 Standardization Efforts

Despite the substantial volume of data generated by onboard sensors and the inherent limitations posed by the dedicated bandwidth of the V2X communication network, the exchange of perception information among CAVs and with the infrastructure has emerged as an evolving research area within C-ITS. One proposed system in this context is CarSpeak, introduced in [68]. CarSpeak represents an innovative approach that empowers CAVs to share sensor information in the form of 3D point clouds. In this system, raw perception data is encoded using an octree scheme, enabling efficient data compression and representation. CAVs then broadcast their associated regions of interest over the network, allowing for the exchange of critical information about the surrounding environment.

In addition, the authors in [69] present a multimodal cooperative perception system that leverages massive vision-based data from camera and Lidars sensors to provide drivers with augmented reality-like see-through views. By processing this information, this system offers a more comprehensive understanding of the road environment. However, experiments in these works have faced critical challenges related to the exchanging of raw perception information which can significantly strain network performance due to the sheer data volume and real-time requirements.

To address these network constraints while preserving the benefits of CP, [70] has proposed an approach that revolves around periodically exchanging only descriptions of tracked objects rather than transmitting raw sensor data. By focusing on essential object-level information, this approach strikes a balance between communication efficiency and the need for comprehensive perception sharing. Several standardization initiatives of CP, such as SAE J3224 of North America [71], CSAE 157 of China [72] and [73], and ETSI TS 103 324 of Europe [37], are existing in the state-of-the-art to meet these challenges.

As this thesis focuses only on the ETSI, the CP standardization is presented by a CPM format and set of message generation rules, as detailed in the following sections [37].

#### 3.3.2.1 CPM Format

The CPM format, as defined by the ETSI, is a structured framework that encompasses various elements to facilitate the exchange of perceived information among ITS-Ss, principally CAVs and RSUs. This format, as depicted in Figure 3.6, includes an ITS Packet Data Unit (PDU) header and four key types of containers: the Management Container, the Station Data Container (SDC), one or more Sensor Information Containers (SICs), and one or more Perceived Object Containers (POCs). The ITS PDU header, as specified in [28], incorporates essential Data Elements (DEs) such as the protocol version, message ID, and station ID, providing essential contextual information. The Management Container (MC) is a mandatory component within the CPM, offering fundamental data about the originating ITS station, including its precise position and type. The optional SDC provides additional information about the originator, distinguishing between CAV and RSU and specifying their unique properties. The SIC is also optional and provides detailed insights into the sensor properties, aiding the receiving ITS stations in understanding the originator's sensing capabilities. Lastly, the optional

POC delves into the dynamic state and properties of detected objects, offering invaluable data for tracking and decision-making.

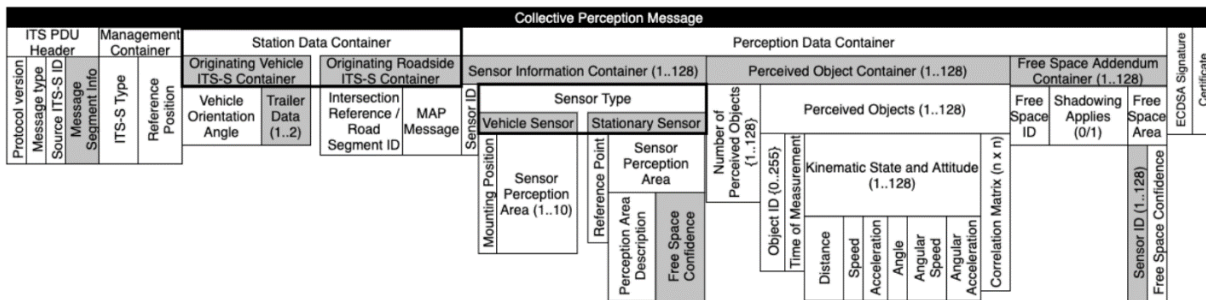


Figure 3.6 CPM format [74]

### 3.3.2.2 CPM Generation Rules

These rules determine when and what information a transmitting ITS-S shall include in the current CPM for transmission. Two policies are under consideration by the ETSI standardization process: the periodic policy and the dynamic policy. The periodic policy mandates that CPMs be generated at regular intervals, denoted as  $T_{GenCpm}$ , and requires the inclusion of information on all primary components of the CPM, even if no objects are detected. This policy serves as a baseline for performance comparison against more advanced policies.

In contrast, the dynamic policy involves the transmitting ITS-S checking the environment every  $T_{GenCpm}$  to decide if a new CPM should be generated and transmitted. A new CPM is generated if a new object is detected or if specific conditions are met for any previously detected objects. These conditions include a change in absolute position by more than 4 m since the last inclusion in a CPM, a change in absolute speed by more than 0.5 m/s, or if the object was included in a CPM one second ago. All newly detected objects and those meeting these conditions are included in the CPM. Even if no object satisfies these conditions, a CPM is still generated every second, including only the MC, SDC, and SICs and excluding the POC. It is essential to note that these CPM generation rules, adapted from CAM generation rules for detected objects, are preliminary proposals subject to potential changes in the future specifications, necessitating careful analysis to comprehend their implications on road traffic and communication.

### 3.3.3 Simulation Tools

This section delves into the simulation tools and perception models employed to simulate and evaluate CP-related works.

### 3.3.3.1 Artery and SUMO

To ensure a realistic evaluation of the proposed approaches, we have chosen a simulation platform that combines the capabilities of both the Artery [75] and SUMO [76] simulators. Specifically, Artery, an extension of Vehicles in Network Simulation (Veins) [77], uses modern C++ modules to simulate the ITS-G5 communication standard for each node which may be a CAV or RSU. On the other hand, SUMO provides detailed modeling of vehicular traffic, capturing the mobility of vehicles across various classes and dimensions, based on predefined roadmaps, traffic signals, and other infrastructural elements. The seamless integration of Artery with SUMO through the TraCI interface ensures a synchronized simulation of traffic dynamics and vehicular communication, providing a robust platform for evaluating the proposed approaches.

### 3.3.3.2 Perception in Artery

The detection of road objects within Artery is realized through a sophisticated modular perception system. Central to this system are the following two principal components:

- **Global Environmental Model (GEM):** Provides a full database that contains information about all nodes within the simulation environment, effectively mapping out a global overview of dynamic objects. This mapping is facilitated through the instantiation of Global Environmental Objects (GEMOs) for each object. These GEMOs are dynamically updated in real-time from SUMO via the TraCI interface, reflecting any changes in their dynamic properties as the simulation progresses. CAVs and RSUs within the perception system are equipped with local perception sensors. These sensors can be configured independently in terms of their perception range, opening angle, and specific mounting points on the CAV or the RSU. The GEM plays a critical role in identifying whether other vehicles fall within the direct line of sight of these sensors, enabling the detection of potential obstructions caused by other vehicles or static obstacles that might impede the sensor's FoV.
- **Local Environmental Model (LEM):** Upon the detection of objects by the local perception sensors, each CAV and RSU creates its own LEM instance, acting as a local database that stores and manages information on all objects within the sensor's perception range. This includes the creation and updating of Local Environmental Model Objects (LEMOs) for each detected object, ensuring that the simulation maintains an accurate and up-to-date representation of the local vehicular environment.

Further elaboration on these components and their interaction within the simulation framework can be found in [78] and [79].

## 3.3.4 Augmented CP

Augmented CP represents a significant advancement in enhancing the perception of the driving environment. This section introduces its definition and scope, essential components, and integration architectures within C-ITS.

### 3.3.4.1 Key Components and Technologies

The foundation of Augmented CP in CAVs is built upon key components and technologies that work in tandem to transform raw data into actionable insights, as depicted in Figure 3.7. The main components can be identified by the following concepts:

- **Infrastructure-Perception Integration and Cooperative Data Fusion:** The concept of integrating infrastructure into the perception system of CAVs marks a significant advancement in their operational capabilities. By equipping infrastructure elements such as traffic lights and road signs with advanced and high-capacity sensors, these components become active participants in the traffic ecosystem and transmit crucial perception-related information directly to CAVs [80]. This development facilitates a sophisticated process of cooperative data fusion, where CAVs perform a fusion process of data from their own onboard sensors with additional information received from nearby vehicles and these sensor-equipped infrastructure elements [81]. The fusion of data can also occur at various levels. At the low level, it involves the processing and fusion of raw sensor data [60]. At the feature level, the focus shifts to extracting and combining specific features from this data, like the shape or speed of nearby objects [82]. At the high level, the process includes complex interpretations and contextual understanding of the environment, such as identifying objects, recognizing traffic patterns, and predicting potential hazards [83]. For instance, the authors in [84] used a multi-target tracking method [85] based on CAV-onboard Lidars and a voxel clustering algorithm to perceive the surrounding environment. This method fuses tracking information with the perception information of the RSU and other CAVs to generate the trajectory of the target CAV. By incorporating communication from infrastructure-based perception, CAVs achieve an enhanced perception of their surroundings. This ensures a more comprehensive understanding of complex driving scenarios, even under the limitation of onboard sensors and the V2V communication failure that can occur due to the network congestion.
- **Edge and Cloud Computing for perception tasks:** The role of AI and ML cannot be overstated in augmenting the perception of the environment. CAVs require substantial processing resources to analyze the vast amount of data generated by their sensors and V2X communication [86]. Edge computing, where processing occurs onboard the CAV, is essential for minimizing latency and ensuring rapid responses to changing conditions. Additionally, cloud-based computing complements edge processing by providing additional computing power and access to vast data sets for training enhancement.



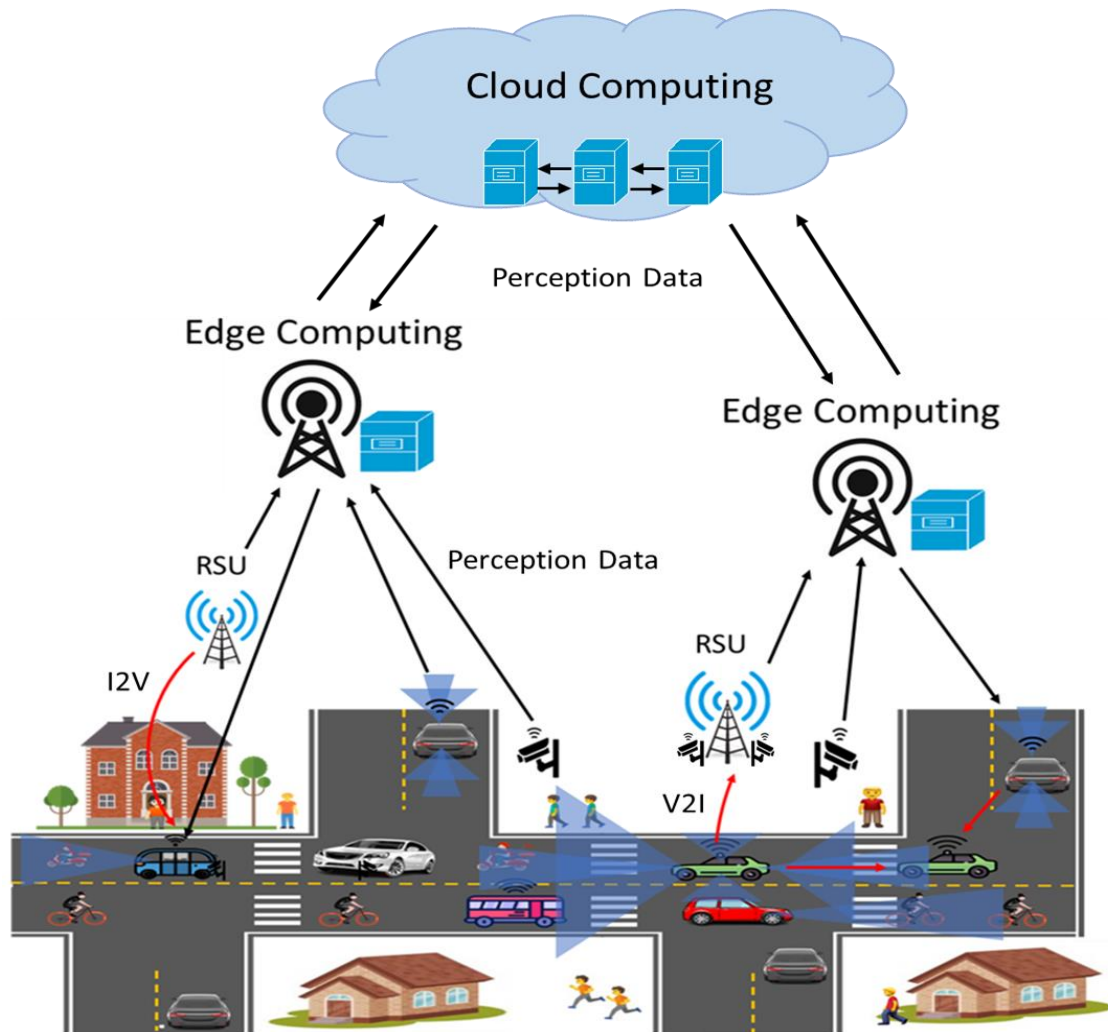


Figure 3.7 Cooperative and augmented perception in C-ITS

### 3.3.4.2 Architecture, Limitations, and Impact on overall perception quality

Figure 3.8 and Figure 3.9 illustrates the CAP's functional architecture for CAVs and RSUs. Both entities perform a local perception module that captures raw data and identifies objects within the environment using onboard sensors. This data is then fed into the CAP module, which can be divided into two principal submodules: augmented perception and CP. The augmented perception submodule receives information from other C-ITS services, the surrounding CAVs, and the infrastructure through V2X communication and then performs a cooperative fusion process to further refine and enhance accuracy and quality. The output from this submodule is concurrently stored in a perception data database to serve the needs of other applications, such as autonomous driving in CAVs and road traffic monitoring in infrastructure. In parallel, this data undergoes a CP process for dissemination in the environment and exchange with edge and cloud platforms for further processing and enhancement.

However, achieving the full potential of CAP and ensuring its operational continuity requires surmounting a series of challenges.

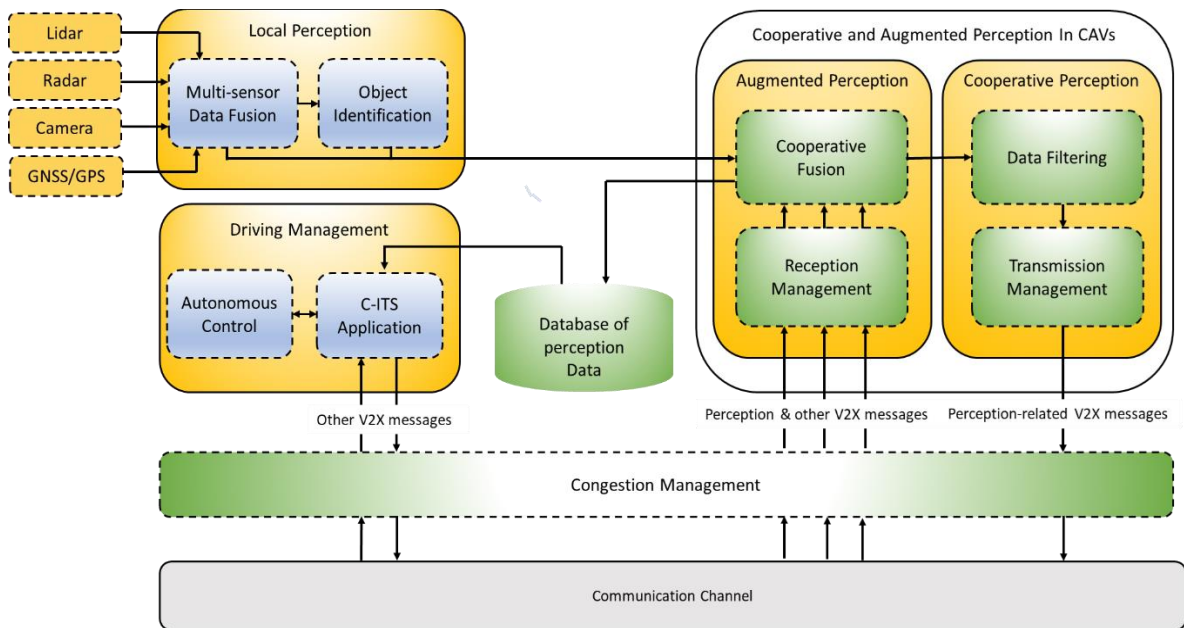


Figure 3.8 Architecture of cooperative and augmented perception in CAVs

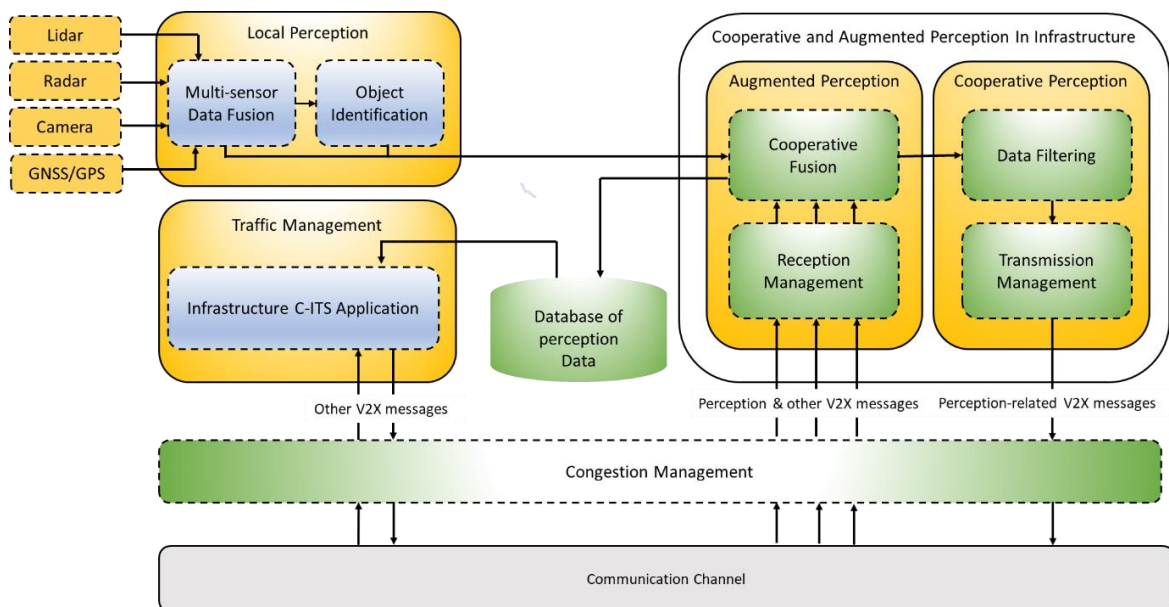


Figure 3.9 Architecture of cooperative and augmented perception in infrastructure

## **3.4 Resilience of the Cooperative and Augmented Perception**

### **3.4.1 Definition and Key Concepts**

#### **3.4.1.1 Definition**

The resilience CAP within C-ITS takes on a specialized definition. It denotes the inherent ability of these systems to adaptively respond to a spectrum of unpredictable conditions commonly encountered in the driving environment. This adaptation is vital for continuously providing other CAVs with reliable and useful perception information, essential for safe operation. The Resilience in such systems is not merely about enduring disruptions but involves a dynamic capability to adjust and respond to various types of changes and challenges. These may include shifts in operational and technical malfunctions, communication resource limitation, network failure, and both internal and external security threats.

#### **3.4.1.2 Key Concepts**

At the core of a resilient systems are several key concepts:

- **Adaptability:** This attribute highlights the system's agility in adjusting its functioning in response to changing circumstances. Adaptability is crucial in environments where sensor inputs, communication links, and operational parameters are subject to variability and unpredictability.
- **Robustness:** This pertains to the system's strength to resist disruptions without losing critical functionality. A robust system maintains its core operations, even under stressful or unexpected conditions.
- **Recoverability:** This feature emphasizes the system's capacity to rapidly recover from disturbances. Effective recoverability ensures that any impact on performance and service quality is minimized and short-lived, enabling a swift return to optimal functioning.

### **3.4.2 Area of Challenges and Resiliency Requirements**

Ensuring resilience involves meeting a multitude of challenges and specific resiliency requirements. A primary challenge involves addressing the increasing volume of data generated by onboard sensors in CAVs, RSUs, and received via V2X communications. This data, potentially comprising imprecise, irrelevant, and redundant information, can saturate communication networks and causes increased congestion. This limitation highlights the necessity for efficient data filtering strategies.

Such strategies must not only efficiently manage the high volume of data but also adapt dynamically to the diverse complexities of the driving environment, meet the specific needs of

receiving CAVs for relevant information, and the changing conditions of network congestion and control mechanisms. To ensure a resilient CAP system, it is crucial that CAVs receive timely and relevant data for safe navigation, even under evolving network failures and interference.

## **3.5 Conclusion**

In summary, this chapter has established a robust foundation for comprehending CAP intricacies and advancements within C-ITS. The insights gained provide essential context for further exploration and innovation in this continually evolving domain. Addressing the array of challenges and bolstering the resilience of CAP systems is pivotal in advancing the capabilities and safety of CAVs across a spectrum of adverse cooperative driving conditions. Among these challenges, ensuring reliable communication emerges as the paramount factor in achieving overall the resilience of CAP. The efficacy of this concept for CAVs fundamentally depends on the consistent and reliable sharing of perception information, effectively expanding the limited horizons of onboard sensors and increasing situational awareness.

# Chapter 4

## Multi-Agent Deep Reinforcement Learning for Object Redundancy Mitigation in V2V Networks

### 4.1 Introduction

Managing the vast amount of data produced by multiple sources, including onboard sensors and network cooperation, has become a major challenge to overcome, as it impedes efficient data exchanging and processing within C-ITS. The authors in [87] and [88] have examined the capacity requirements for communication among CAVs across varying traffic densities and market penetration rates. This study reveals that, when the number of CAVs increases, the amount of perception data to share also increases significantly due to the dynamic nature of the driving environment.

To meet this challenge, ETSI has introduced the CPM format and message generation rules [37]. This CP standardization is designed to strike a balance between limited channel capacity and the increasing volume of perception data required in vehicular communications. However, while CAVs do not actively analyze perceived information from their onboard sensors and neighboring CAVs, the process of CP often leads to a substantial amount of redundant information being transmitted in the network when using these generation rules [89]. This redundancy occurs when multiple CAVs, each equipped with its onboard sensors, simultaneously perceive and share overlapping data concerning the same objects within their environment. While this redundancy underscores the reliability of the data, it also presents various challenges related to data overload, network congestion, and processing inefficiencies. These challenges significantly impair the quality of CAP, particularly in scenarios marked by high vehicle density, congestion, and resource-limited networks.

The relevance of perception data highly depends on what the receiving CAVs can perceive. In such scenarios, specific data types can be vital for ensuring safety and efficient navigation. For instance, in congested urban environments, data characterizing the blinded stops of nearby CAVs is crucial for transmission to make informed decisions. However, the same data might be useless in less complex environments, such as highways, where traffic dynamics are more predictable and the perceptual challenges less complex. Consequently, there is an urgent need for sophisticated RMMs that can adapt to the varying contexts of perception. These RMMs

should possess the capability to intelligently discern and omit unnecessary data from the CPM, thereby maintaining the efficiency and reliability of CAP.

This chapter aims to address this need by exploring methods to intelligently mitigate data redundancy. Its core foundation centers on the utilization of MDRL to develop, V2V-RMM, an intelligent method to mitigate perception data redundancy in the V2V network without decreasing the environmental awareness about objects for CAVs. V2V-RMM's effectiveness lies in its adaptability to diverse perception contexts. It enables each CAV to learn how to minimize redundant data by maximizing the relevance of information to include in the CPM. This is achieved by leveraging various DRL algorithms in a distributed setting. It involves considering various perception factors, including distance, object dimensions, viewing angles, and occlusions created by caused by other road users. This approach aims to dynamically adapt to the environment's changing conditions and effectively optimize the exchange of CPMs without decreasing object awareness for CAVs.

## 4.2 Related Works

Several RMMs have been proposed to mitigate the transmission of redundant information in the CPMs.

### 4.2.1 ETSI RMMs

This section overrides the RMMs that have already been considered by the ETSI standardization of CP [37].

#### 4.2.1.1 Dynamics-Based

This RMM serves as an initial strategy, following the same logic of the CPM generation rules. It filters an object for inclusion in the CPM if its position or speed changes less than predefined position and speed thresholds  $P\_Redundancy$  and  $S\_Redundancy$  from the last update received about it [90]. This approach adapts the number of updates to each object independently, catering to dynamic scenarios. For instance, an object moving rapidly might receive more frequent updates than a slower-moving one. This method aids tracking algorithms and optimizes the use of communication resources.

#### 4.2.1.2 Distance-Based

Using this RMM, the transmitting CAV omits an object if it has recently been received within a specified distance  $D\_Redundancy$  during a defined time window  $W\_Redundancy$ . For instance, if a CAV within  $D\_Redundancy$  has already communicated information about an object, it becomes redundant for another CAV to transmit the same updates within  $W\_Redundancy$ . However, efficiently determining parameters like the time window and range is crucial, as an excessively small value may lead to insufficient filtering, while an overly high value may result in excessive redundancy.

### 4.2.1.3 Frequency-Based

Filtering objects using this RMM is based on the number of updates received within a specified time window  $W\_Redundancy$ . If this number is equal or higher than a  $N\_Redundancy$  threshold, the object will not be considered for transmission. A benefit of this method is that the size of the CPM can be reduced using  $N\_Redundancy$ , which can reduce the number lost CPM in the network. However, depending on the time window parameter, the reception of all updates in a short interval could block further transmissions for the remainder of the window.

### 4.2.1.4 Entropy-Based

The exchange of object lies in assessing the expected perception of neighboring CAVs within the  $D\_Redundancy$  range and checking whether the novelty of the measured information is less than an entropy threshold  $E\_Redundancy$ , during  $W\_Redundancy$  [91]. The suggested value within  $D\_Redundancy$  is to be lower or equal to the typical communication range. However, it introduces computational overhead and complexity, with the anticipated knowledge relying on fusion algorithms and metrics, making the estimation more challenging.

### 4.2.1.5 Confidence-Based

This RMM filters an object based on the maximum confidence level of object information received by other CAVs compared to the confidence of the local perception of the transmitting CAV. This strategy prioritizes transmission when, during  $W\_Redundancy$ , the estimated knowledge about an object is higher than the confidence of the received object by CPM. The challenge lies in defining confidence, as it can vary significantly between CAVs, based on onboard sensors and sensor-fusion algorithms.

An evaluation of these RMMs highlighted that controlling redundancy positively impacts network metrics, such as CBR and packet error rate, without reducing the detection ratio [92]. Still, a significant limitation of ETSI's RMMs is their reliance on static thresholds, which may be unsuitable in dynamic traffic conditions with varying densities. Furthermore, these strategies assume uniform redundancy control methods among all vehicles, an assumption that might not hold in mixed traffic where some vehicles cannot send and receive messages.

## 4.2.2 Further Enhanced RMMs

Authors in [93] have proposed redundancy control methods based on channel status, the number, and the type of V2X stations providing similar perceived information. The primary goal is to adjust the number of V2X stations transmitting data about the same object according to the channel load, while keeping the CP awareness close to the standard CPM generation rules. However, these methods also rely on predefined and static thresholds, which may not be suitable in diverse driving environments characterized by variable situations and vehicular densities. Furthermore, [94] demonstrated that current CPM generation rules could lead to substantial redundancy in highway scenarios. They theoretically proposed a probabilistic data

selection scheme [95], which enables each CAV to adjust its adaptive transmission probability for each detected object, considering its position and traffic information. Yet, the communication aspect of this method was tested using MATLAB instead of network simulators like Artery, which emulates communications using the ETSI ITS-G5 protocol stack. Generally, most existing techniques employ predefined thresholds to determine whether to include perceived objects in a CPM, based on criteria like position or speed. However, there is still a lack of consideration for the actual usefulness of this information, relative to the coverage area of the transmitting CAV.

Moreover, DRL techniques can be applied to vehicular communications to optimize data redundancy during transmission in the network. DRL enables agents to learn optimal policies through interactions with their environment, aiming to maximize cumulative rewards [96]. The study [97] introduced a DRL-based CP (DRLCP) method, where each CAV intelligently selects information for inclusion in the CPM, based on the context perceived by its onboard sensors. This method focuses on allocating communication resources to critical packets and minimizing redundancy. However, this method was evaluated on only two road networks and did not consider the impact of obstructions on preserving perception awareness for CAVs.

Regarding this limitation, the work in [98] has proposed omitting duplicated perceived objects with objects that neighbor CAVs can perceive. Based on that RMM, CAVs exchange perception information depending on an empty, occupied, or occluded state of the grid-based projection of their FoVs. However, these schemas broadcast CPMs without considering object usefulness over the network coverage of each transmitting CAV. This still represents an open challenge in maintaining perception awareness, especially in highly congested networks. The work presented in [98] has proposed omitting redundant objects based on object usefulness. The authors of this article have introduced a deep RL-based schema where object usefulness is modeled as a reward based on only the distance from the perceived object to the CAV receiving it. Nonetheless, this modeling of object usefulness still lacks the consideration of different perception contexts such as object size and road occlusions, which may considerably affect the quality of CAP. In the section below, we propose the formulation of object usefulness as a maximization problem by considering multiple perception contexts such as position, distance, object size, viewing angle, and occlusions caused by other road users.

### **4.3 Problem Formulation**

In this section, we aim to establish a comprehensive framework for evaluating the usefulness of information obtained through onboard sensors and V2V communication within a multi-CAV environment. Our approach formulates usefulness as a maximization problem, where the goal is to optimize the value of exchanged perception data and reduce network load, thereby improve the communication reliability. This allows CAVs to increase their perception awareness by exchanging further useful information that may be needed for safe operations. The formulation is intricately designed to account for a variety of perceptual contexts, including, but not limited to, position, distance, object size, viewing angle, and potential occlusions.



### 4.3.1 Architecture

Figure 4.1 illustrates the principal internal services pertinent to our study for a given CAV.

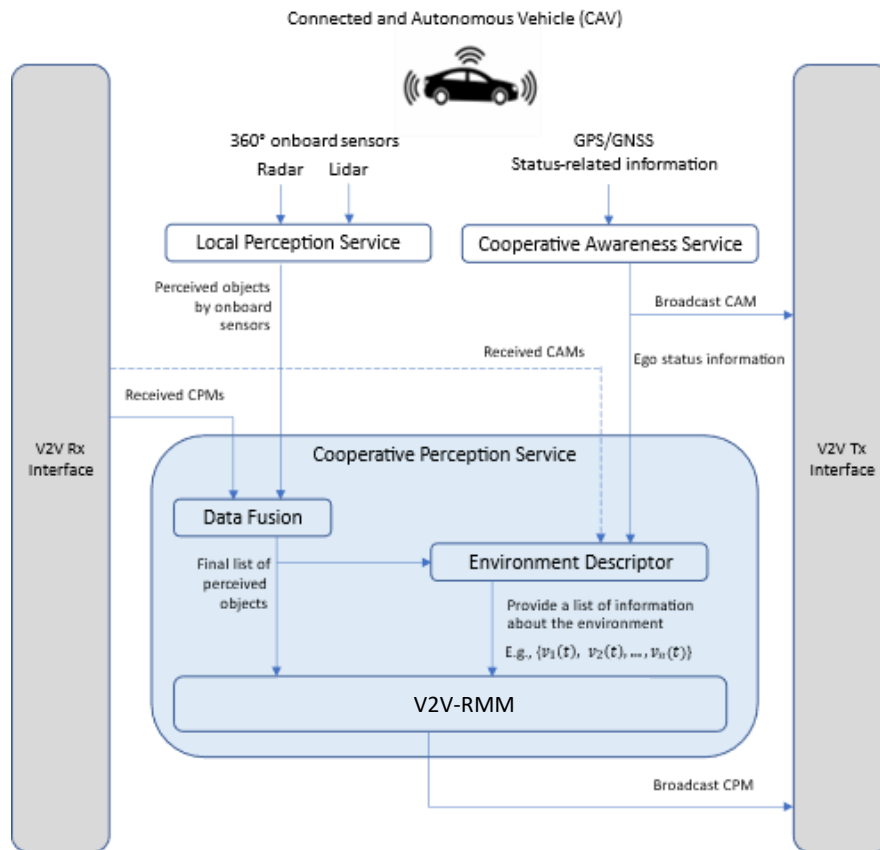


Figure 4.1 Basic services used by each CAV.

This architecture encompasses:

- Local Perception Service (LPS): Utilizes the onboard sensors to perceive the road environment, outputting a list of detected objects.
- Cooperative Awareness Service (CAS): This service gathers information from GPS/GNSS and other relevant sources to generate and disseminate status information via CAMs to other CAVs through the V2V Transmission (Tx) Interface.
- Cooperative Perception Service (CPS): Allows CAVs to send and receive objects with each other via CPMs. The core of this service encompasses three principal modules
- Data Fusion Module (DFM): Fuses locally perceived objects with received objects from other CAVs through the V2V Rx Interface. This module outputs the final list of objects to be included in a CPM and broadcasted to other CAVs.
- Environment Descriptor Module (EDM): Manages data from onboard sensors, CAMs, and CPMs to build a comprehensive view of the driving environment.

- V2V-RMM Module: V2V-RMM module takes the description of the environment as well as the perceived objects as inputs and generate an optimized CPM with mitigated redundancy for transmission.

### 4.3.2 Optimization Problem

To build optimization problem that V2V-RMM aims to solve, we begin by freezing the image of the driving environment at a specific awareness timestep,  $t_{av}$ , as illustrated in Figure 4.2. This environment includes multiple vehicles, denoted by  $V = \{v_1, v_2, \dots, v_n\}$ , varying in dimensions, classes, types, and driving behaviors. Additionally, the environment includes static road obstacles like buildings and trees. In our model, vehicles are simplified as rectangles, each characterized by  $v(t_{av}) = \{c(t_{av}), l, w\}$ , where  $c$  represents the geometric center with X and Y coordinates  $(x(t_{av}), y(t_{av}))$ , and  $l$  and  $w$  denote length and width, respectively.

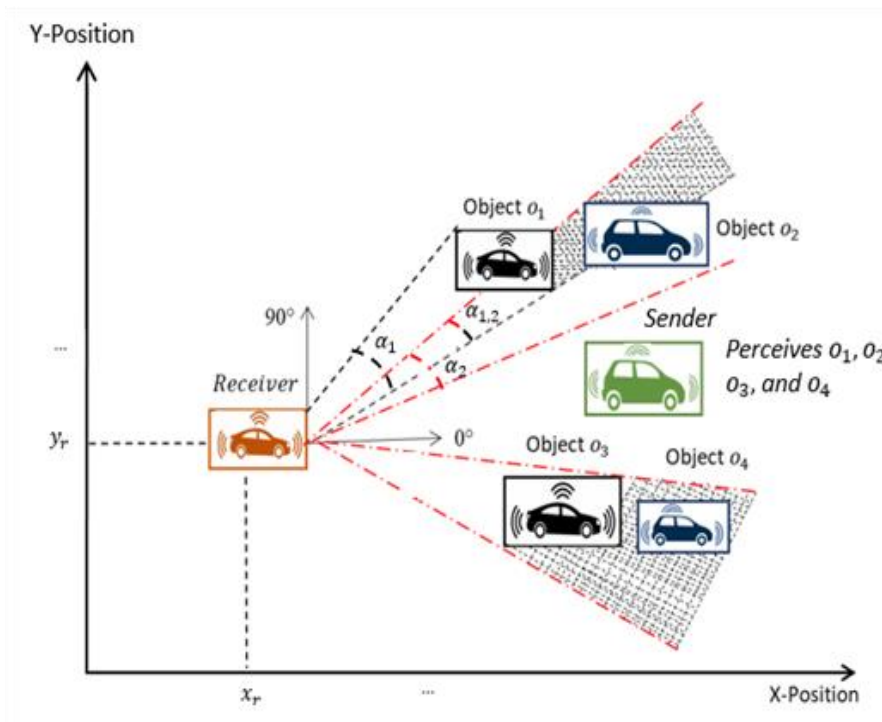


Figure 4.2 An illustration of road occlusions in the driving environment

Each designated CAV of index  $i$  is represented by a set of features including its position, speed, length, and width. For the purposes of this study, we focus on three key attributes: position on a global 2-dimensional plane, length, and width, as defined in equation (4.1):

$$h_i(t_{av}) = \{x_i(t_{av}), y_i(t_{av}), l_i, w_i\} \quad (4.1)$$

Following the exchange of CAMs, every CAV receives the status of other CAVs in its communication range through the V2V Rx Interface. These statuses can be denoted by equation (4.2):

$$H_i(t_{av}) = \{h_j(t_{av}), j \neq i\} \quad (4.2)$$

We consider another timestep,  $t_{pv}$ , when each CAV of index  $i$  performs CPS. The final list of objects provided by DFM can be denoted by a list of perceived features, such as position, length, and width for each object  $o$ , as shown in equation (4.3):

$$T_i(t_{pv}) = \{o_{i,1}(t_{pv}), o_{i,2}(t_{pv}), \dots, o_{i,k}(t_{pv})\}, \quad (4.3)$$

with each perceived object from this list is defined by:

$$o_{i,i}(t_{pv}) = \{x_i(t_{pv}), y_i(t_{pv}), l_i, w_i\} \quad (4.4)$$

The main role of V2V-RMM is to enable each CAV to select and broadcast objects that maximize the relevance for surrounding CAVs, effectively reducing redundancy. The objective function for maximization is denoted by:

$$\underset{P_i(t_{pv})}{\text{maximize}} \quad 1 - \left( \frac{1}{n_i * n_o} \sum_{r \in C_v(t_{pv})} \sum_{o \in P_i(t_{pv})} f_{o,r}(t_{pv}) * g_{o,r}(t_{pv}) \right), \quad (4.5)$$

Subject to:

$$t_{av} < t_{pv} < 2 * t_{av}, \quad (4.6)$$

$$P_i(t_{pv}) = \{o_{i,1}(t_{pv}), o_{i,2}(t_{pv}), \dots, o_{i,n_o}(t_{pv})\} \subseteq T_i, \quad (4.7)$$

Where  $P_i$  is the list of  $n_o$  relevant objects to be included in the current CPM.  $n_v$  is the number of CAVs within the communication range  $C_i$  of the transmitting CAV  $i$ , at time  $t_{pv}$ . The selection process dynamically adapts to the driving environment situation without employing static thresholds, using information provided by the EDM. This module creates a geometric representation of surrounding elements at each timestep  $t_{pv}$  (e.g., provides as an output a list of the position, length, and width for each surrounding element). Equation (4.6) ensures that CAVs broadcast and receive CAMs and update their status before sharing CPMs.

$$f_{o,r}(t_{pv}) = \begin{cases} 0, & d_{o,r}(t_{pv}) > m \\ 1 - \frac{d_{o,r}(t_{pv})}{m}, & \text{otherwise} \end{cases} \quad (4.8)$$

$$g_{o,r}(t_{pv}) = \begin{cases} 0, & \phi_{o,r}(t_{pv}) > \rho_{o,r}(t_{pv}) \\ 1 - \frac{\phi_{o,r}(t_{pv})}{\rho_{o,r}(t_{pv})}, & \text{otherwise} \end{cases} \quad (4.9)$$

The relevance of objects within the detection range of a receiving CAV is significantly influenced by their relative positions and potential obstructions that may obstruct the LoS of its onboard sensors. Equation (4.8) accounts for this by having each transmitting CAV compute distance and occlusion-related factors for each object concerning all possible receivers within its communication range. The distance-related factor, denoted as  $f_{o,r}$  ranges between 0 and 1, signifying the extent to which object  $o$ , perceived by the transmitting CAV, appears within the sensing coverage of receiving CAV  $r$ . Specifically, as the Euclidean distance  $d_{o,r}$  between the object and the receiver increases and gets closer to  $m$ ,  $f_{o,r}$  decreases to zero, indicating that  $o$  is becoming less perceptible, therefore, more useful for this receiving. The distance-related factor decreases to zero, indicating that  $o$  becomes less perceptible and, therefore, more useful for  $r$ . Meanwhile, the occlusion-related  $g_{o,r}$  is denoted by Equation (4.9). This factor ranges also between 0 and 1, and denotes the degree to which  $o$  is directly within the LoS of  $r$ . This factor is proportional to the sum of angles that overlap and occlude the viewing angle  $\rho_{o,r}$  from  $r$  to  $o$ . The closer this sum is to the observation angle, the more the object's LoS is obscured. For instance, as depicted in Figure 4.2, the sender detects objects  $o_1, o_2, o_3$ , and  $o_4$ . For the receiver, the angle  $\alpha_{1,2}$  represents the occlusion angle caused by object  $o_1$ , obstructing its viewing angle to the object  $o_2$ . To calculate these angles, we use the *atang2* function, which computes the angle between two positions in a global 2D coordinate system. Since *atang2* calculates the angle between a given position and the X-axis, a simple subtraction is performed to find the required angle.

## 4.4 System Model and Learning Algorithms

This section delves into the modelling of the proposed approach and learning algorithms used to solve the optimization problem formulated in the previous section.

### 4.4.1 Motivation on DRL

ML and Deep Learning (DL) offer sophisticated solutions to a wide array of complex problems across numerous domains. At the core of these technologies is their ability to process and learn from large datasets, enabling them to identify patterns, make predictions, and automate decision-making processes. DL, in particular, excels in handling high-dimensional data, thanks to its neural network architectures. However, these technologies heavily depend on the availability of large and well-labeled datasets. In many real-world scenarios, especially CAP in driving environments, such data may not exist or be extremely costly and time-consuming to generate. Furthermore, ML and DL models often require significant computational resources for training and can be prone to overfitting, especially when the data does not represent the full spectrum of the problem space. Additionally, they typically offer little adaptability to dynamic environments where the underlying data patterns can change.

The use of RL can address many of these constraints. Unlike traditional ML approaches, RL does not rely on pre-existing datasets. Instead, it learns by interacting with an environment, making decisions, and observing the outcomes of these actions through rewards. This approach

is particularly beneficial in solving combinatorial optimization problems, where the solution space is large and dynamic.

Combing the of power of DL with RL (DRL) to continuously learn and adapt its strategies based on the reward from the environment makes it an ideal approach for tasks where the optimal decision-making policy is not known in advance or is subject to change for adaptability. Its exploration-exploitation trade-off allows DRL to discover novel solutions which may not be drawn from datasets using ML, DL and even RL traditional approaches in complex and multi-dimensional spaces, rendering it more flexible and robust in environments with high uncertainty and variability.

To that end, we leverage the ability of RL to learn effective strategies for object selection and broadcasting in V2V networks, mitigating the transmission of redundant information by adapting to various perception contexts, as described above. This allows for improving the reliability of the communication network by reducing channel overload, which enables CAVs to exchange more relevant information, ultimately leading to an improvement in the quality of their CAPs.

## 4.4.2 System Model

Agents in MDRL interact with a stateful environment to solve a sequential decision-making problem. Each agent acts based on the state of the environment and then receives feedback at each timestep to maximize its reward. Therefore, we define the following RL essential components: environment state, action, and reward as follows.

### 4.4.2.1 Environment State

As CAVs move within the cooperative driving environment and collect experienced state-action to improve their future decisions, a position-based representation cannot be used to conduct the training process. Therefore, we introduce a flexible position-independent representation of the environment state that allows a transmitting CAV  $i$  to draw on its past experiences in various locations at any given time, as follows:

$$s_i(t_{pv}) = \{(d_{i,j}(t_{pv}), \beta_{i,j}(t_{pv}), l_j, w_j); \forall j \neq i\}, \quad (4.10)$$

where  $d_{i,j}$  and  $\beta_{i,j}$  are the distance and viewing angle from the transmitting CAV  $i$  to each possible receiving CAV  $j$  in its communication range.  $l_j$  and  $w_j$  are the length and width of CAV  $j$ .

### 4.4.2.2 Action Space

Given the mobility of CAVs in the driving environment, the perception changes over time. Therefore, the action should be independent of changing characteristics. To that end, we propose a cell-based scheme that divides the circular FoV of each CAV into  $p$  pistes and  $s$

sectors. Therefore, the FoV of the CAV  $i$  at  $t_{pv}$  can be represented by a vector of cells of size  $s * p$  as follows:

$$FoV_i(t_{pv}) = [Ce_{1,1}^0(t_{pv}), \dots, Ce_{1,s-1}^1(t_{pv}), Ce_{1,0}^2(t_{pv}), \dots, Ce_{p-1,s-1}^{p*s-1}(t_{pv})], \quad (4.11)$$

where  $Ce_{p',s'}^j$  is representing the cell of  $p'th$  piste and  $s'th$  sector indexed by  $j \in [0, p * s - 1]$ . To that end, we define the action space by the power set of  $FoV_i$  in order to cover all the unique possible combinations of cells. The action  $a_i$  can be a natural number in the range  $[0, 2^{p*s} - 1]$ . This number is then converted to a binary string  $(b_0 b_1 \dots b_{p*s-1})_2$  of length  $p * s$ . Following that, the objects that appear in  $Ce_{p',s'}^j$  will be included in CPM only if  $b_j = 1$ . The CPM include all perceived objects if all bits are set to 1s. On the other hand, the CPM is empty if all bits are set to 0s.

#### 4.4.2.3 Reward

The reward of CAV  $i$ , denoted as  $r$ , is calculated based on the selected action's outcome set of objects  $M_i$  at time  $t$ :

$$r_i(t) = 1 - \left( \frac{1}{n_i * n_o} \sum_{j \in C_i(t)} \sum_{o \in M_i(t)} f_{o,j}(t) * g_{o,j}(t) \right), \quad (4.12)$$

### 4.4.3 Learning Algorithms

The goal of V2V-RMM is to find a CPM content selection policy that maximizes the relevance of objects for the receiving CAVs, thereby reducing the transmission of redundant information in the V2V network. Considering a discrete time step  $t$ , each transmitting CAV  $i$  builds the state of its environment  $s_i(t)$  performs an action,  $a_i(t)$ , based on its strategy,  $\pi_i(s_i(t), a_i(t))$ , then obtains a reward  $r_i(t)$ . The environment then passes from the current state  $s_i(t)$  to the next state  $s_i(t + 1)$ . The cumulative reward can be defined by  $R_i(t) = \sum_{i=0}^t \gamma^{t-i} r_i(t)$ , where  $\gamma \in [0,1]$  is a discount factor given to previous rewards to decrease their influence on the current value. Various learning algorithms are proposed to adjust this strategy, improve the agent behavior and maximize the cumulative reward.

#### 4.4.3.1 Q-Learning

The Q-Learning [100] can be used to learn a CAV  $i$  how select optimal actions in a given driving environment by learning an action-value function, known as Q-function. The Q-function, denoted by  $Q_i^\pi(s_i(t), a_i(t))$ , represents the expected value of taking an action  $a_i(t)$  in the state  $s_i(t)$  by following its data selection strategy  $\pi_i$ . The goal of Q-learning is to learn the optimal policy  $\pi^*$  that maximizes the total reward over time.

The core of the Q-learning algorithm is the Q-function update rule, which uses the Bellman equation to iteratively update Q-values towards their optimal values. After taking an action  $a_i(t)$  in the state  $s_i(t)$  and observing the reward  $r_i(t)$  and the next state  $s_i(t + 1)$ , the Q-value for  $s_i(t)$  and  $s_i(t + 1)$  is updated as follows:

$$Q_i^\pi(s_i(t), a_i(t)) = Q_i^\pi(s_i(t), a_i(t)) + \alpha \left[ r_i(t) + \gamma \max_{a_i(t+1)} Q_i^\pi(s_i(t+1), a_i(t+1)) - Q_i^\pi(s_i(t), a_i(t)) \right], \quad (4.13)$$

where  $\alpha$  is the learning rate, controlling the extent to which the newly acquired information overrides the old information.  $\gamma$  is the discount factor, representing the importance of future rewards compared to the current rewards.  $\max_{a_i(t+1)} Q_i^\pi(s_i(t+1), a_i(t+1))$  is the maximum predicted reward obtainable in the next state  $s_i(t+1)$ , representing the best future reward the CAV expects to achieve. The Q-learning algorithm uses a lookup table that stores the state-action value functions of all actions in the action space. This method can be impractical for time-sensitive applications, such as the CP, due to its high consumption of time and memory when dealing with large state and action spaces.

#### 4.4.3.2 Deep Q-Network

DQN [100] is an extension of Q-Learning that integrates DL with RL to handle environments with high-dimensional state and action spaces [101]. Developed by researchers at DeepMind, DQN uses a DNN to approximate the Q-value function,  $Q_i(s_i(t), a_i(t), \theta_i)$ , where  $\theta_i$  represents the weights of the neural network. Given the representation of the state and action spaces in this work, which are based on distances, viewing angles, and FoV cells, respectively, approximating the Q-value using a DNN enables the CAV to generalize across similar states and action pairs. This makes it possible to learn effective policies in complex environments.

Figure 4.3 depicts the internal components of the DQN algorithm for a given CAV  $i$ . As can be seen, the main components of this algorithm are two separate DNNs and an experience replay. These networks are used to stabilize the training process. Meanwhile, the experience replay is used to store transitions collected during exploration to conduct the training of the DNN. Algorithm 1 illustrates the leaning process of V2V-RMM using DQN.

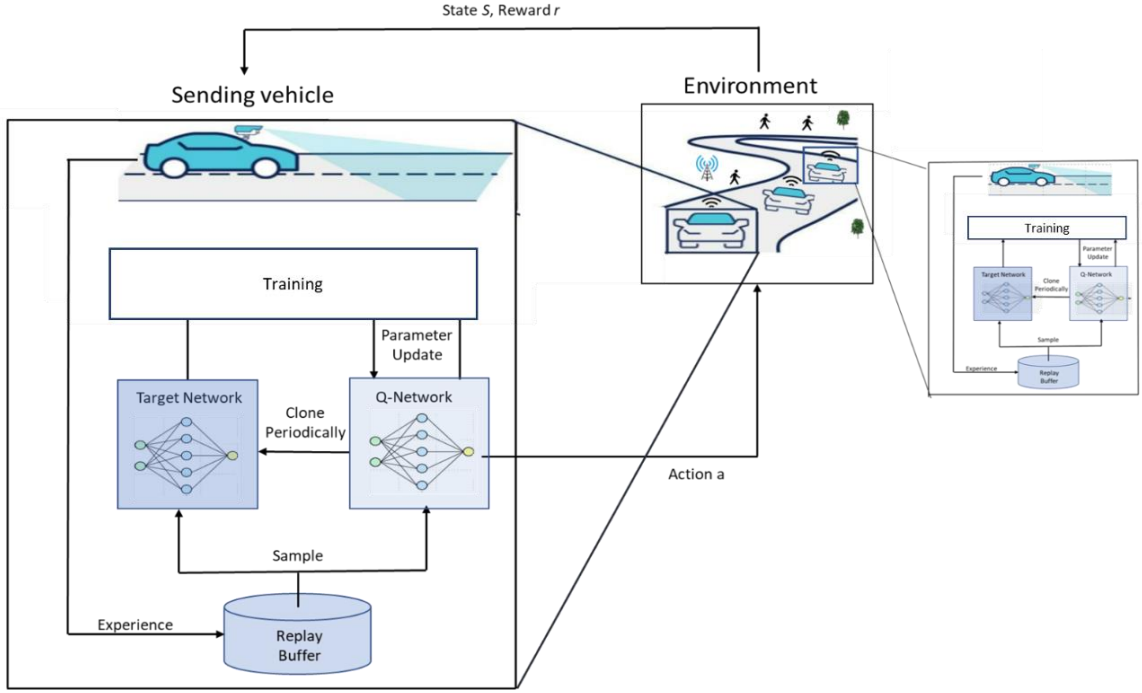


Figure 4.3 DQN basic components internal the CAV

Starting by initializing a replay memory  $D_i$  and two DNN for each CAV  $i$ : the Q-network, which approximates the Q-value function with weights  $\theta_i$ , and the target network, an identical architecture with weights  $\theta_i^-$  initially set equal to  $\theta_i$ . This design choice mitigates the instability arising from rapid successive updates that traditional Q-learning faces when applied to complex non-linear function approximators like DNNs [102]. The core of DQN revolves around each CAV interacting with its environment in discrete time steps, using an  $\epsilon$ -greedy manner. This manner is used to achieve a balance between exploration and exploitation by choosing random actions with probability  $\epsilon$  or actions with the highest estimated Q-value otherwise. Following each CAV  $i$  selects an action, it stores the transition  $e_i(t) = (s_i(t), a_i(t), r_i(t), s_i(t+1))$  in its replay buffer  $D_i = [e_i(0), e_i(1), \dots, e_i(t)]$ . The train network is updated by minimizing the squared loss, which is calculated from a randomly selected mini-buffer from  $D_i$ , as follows:

$$\text{loss}(\theta_i) = E_{(s,a,r,s') \sim U(D_i)} \left[ \left( y_i^{DQN} - Q_i(s, a, \theta) \right)^2 \right], \quad (4.14)$$

where

$$y_i^{DQN} = r_i' + \gamma \max_{a_i'} Q_i(s_i', a_i', \theta_i^-), \quad (4.15)$$

and  $s_i'$ ,  $a_i'$ , and  $r_i'$  are the state, action, and reward of the next time step.



### 4.4.3.3 Double DQN

DDQN is an advancement of the traditional DQN algorithm, addressing its tendency to overestimate action values under certain conditions. This overestimation arises from the max operation used in the Q-learning target network, where the same values are both selected and evaluated, leading to optimistic value estimates [103].

The key innovation behind the DDQN algorithm is to decouple the selection of the action from its evaluation. The learning process follows the same as DQN described above, however, its consider the following loss calculation:

$$loss(\theta_i) = E_{(s,a,r,s') \sim U(D_i)} \left[ \left( y_i^{DDQN} - Q_i(s, a, \theta) \right)^2 \right], \quad (4.16)$$

with

$$y_i^{DDQN} = r_i' + \gamma Q_i \left( s_i', \arg \max_{a_i} Q_i(s_i', a_i', \theta_i), \theta_i^- \right), \quad (4.17)$$

instead of the loss function that uses  $y_i^{DQN}$  to compute the target network. Here, the action  $a_i'$  that maximizes the future reward is selected by the train network  $\theta_i$ , but the evaluation of its value is performed using the target network  $\theta_i^-$ .

---

**Algorithm 1:** The DQN algorithm for learning V2V-RMM

---

```
1: Inputs:
2:  $N_{episodes}$ : number of episodes
3:  $N_{steps}$ : number of learning steps per episode
4:  $N_{init}$ : number of episodes to re-initialize the target network
5:  $\epsilon$ -greedy  $\epsilon \in ]0,1[$ 
6: Output:
7: An optimal object selection strategy
8: Begin:
9: Initialize a buffer memory,  $D_i$ , for each CAV  $i$ 
10: Initialize two DNNs,  $\theta_i, \theta_i^- \leftarrow \theta_i$ 
11:  $episodes \leftarrow 0$ 
12: While  $episodes < N_{episodes}$  do
13:    $steps \leftarrow 0$ 
14:   While  $steps < N_{steps}$  do
15:     Each CAV  $i$ :
16:       Builds  $s_i(steps)$  based on (4.10)
17:       Samples a number,  $c$ , from the uniform (0,1)
18:       if  $c \leq \epsilon$  then
19:         Predicts  $a_i(steps)$ , a possible combination of cells in its FoV defined by
20:         (4.11)
21:       else
22:          $a_i(steps) = \arg \max_{a_i(steps)} Q_i(s_i(steps), a_i(steps), \theta)$ 
23:       End if
24:       Obtains  $r_i(steps)$  and observes  $s_i(steps + 1)$ 
25:       Stores  $e_i(steps) \leftarrow (s_i(steps), a_i(steps), r_i(steps), s_i(steps + 1))$  in  $D_v$ 
26:      $steps \leftarrow steps + 1$ 
27:   End while
28:   Each CAV  $v$ :
29:     Samples a minibatch from  $D_i$ 
30:     Performs a gradient descent on  $loss(\theta_i)$  (4.14) and updates  $\theta_i$ 
31:    $episodes \leftarrow episodes + 1$ 
32:   After each  $N_{init}$  episodes, each CAV  $i$  resets its target network by the train network
33:    $\theta_i^- \leftarrow \theta_i$ 
34: End while
35: End
```

---

---

**Algorithm 2:** The DDQN algorithm for learning V2V-RMM

---

```
1: Inputs:
2:  $N_{episodes}$ : number of episodes
3:  $N_{steps}$ : number of learning steps per episode
4:  $N_{init}$ : number of episodes to re-initialize the target network
5:  $\epsilon$ -greedy  $\epsilon \in ]0,1[$ 
6: Output:
7: An optimal object selection strategy
8: Begin:
9: Initialize a buffer memory,  $D_i$ , for each CAV  $i$ 
10: Initialize two DNNs,  $\theta_i, \theta_i^- \leftarrow \theta_i$ 
11:  $episodes \leftarrow 0$ 
12: While  $episodes < N_{episodes}$  do
13:    $steps \leftarrow 0$ 
14:   While  $steps < N_{steps}$  do
15:     Each CAV  $i$ :
16:       Builds  $s_i(steps)$  based on (4.10)
17:       Samples a number,  $c$ , from the uniform (0,1)
18:       if  $c \leq \epsilon$  then
19:         Predicts  $a_i(steps)$ , a possible combination of cells in its FoV defined by
20:         (4.11)
21:       else
22:          $a_i(steps) = \arg \max_{a_i(steps)} Q_i(s_i(steps), a_i(steps), \theta)$ 
23:       End if
24:       Obtains  $r_i(steps)$  and observes  $s_i(steps + 1)$ 
25:       Stores  $e_i(steps) \leftarrow (s_i(steps), a_i(steps), r_i(steps), s_i(steps + 1))$  in  $D_i$ 
26:        $steps \leftarrow steps + 1$ 
27:     End while
28:     Each CAV  $i$ :
29:       Samples a minibatch from  $D_i$ 
30:       Performs a gradient descent on  $loss(\theta_i)$  (4.17) and updates  $\theta_i$ 
31:      $episodes \leftarrow episodes + 1$ 
32:     After each  $N_{init}$  episodes, each CAV  $i$  resets its target network by the train network
33:      $\theta_i^- \leftarrow \theta_i$ 
34:   End while
35: End
```

---

## 4.5 Performance evaluation

This section evaluates the performance of V2V-RMM based on simulation using Artery, OMNeT++, and SUMO.

### 4.5.1 Scenario and Parameters

We consider a of a real-world obtained from OpenStreetMap and comprises different road traffic scenarios, including the city center and highways with different situations, such as ramps and T-junctions. On this map, we randomly generate vehicles with different types and sizes. We set up a GPS and 360° radar and lidar sensors for each CAV with a maximum sensing range of 100 m. Each CAV employs CAS and CPS to exchange CAMs and CPMs with other CAVs every 0.1s and 0.15s, respectively, using a 6Mbps data rate. We consider the ETSI ITS-G5 as a V2V communication profile for CAVs with a communication coverage of 500 m, where they can broadcast and receive messages through the control channel (CCH). We implemented V2V-RMM based on the PyTorch library. We define a Multilayer Perceptron (MLP) network on each CAV to represent its DNN approximation functions. The simulation spans 70000 time slots, where each time slot represents a CPM generation interval. The training phase consists of  $N_{ep} = 6000$  learning episodes, where each episode be made up every  $N_{steps} = 10$  CPM generation intervals. Furthermore, we consider the RMSProp optimizer [104] with a learning rate  $\alpha = 10^{-3}$ , a minibatch size of 64, a discount factor  $\gamma = 0.99$ , and a buffer size  $|\mathcal{D}| = 10^6$  for each CAV to conduct the training process.

### 4.5.2 Analysis of Training Convergence

Let MDQN and MDDQN represent the learning algorithms of V2V-RMM based on multi-agent DQN (Algorithm 1) and multi-agent DDQN (Algorithm 2), respectively. Figure 4.4 illustrates the variation of the average reward using each algorithm as a function of the number of episodes. The proposed method using MDDQN shows superior performance in maximizing the average reward over 6000 episodes compared to the MDQN. This suggests that using two separate DNNs for action selection and evaluation in the proposed method allows CAVs to learn more effective policies that optimize the usefulness of objects in V2V networks.

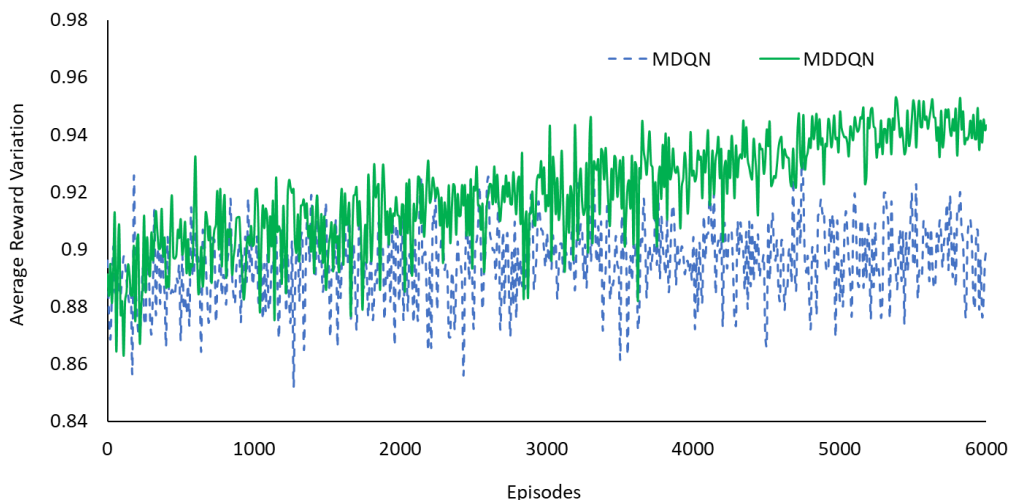


Figure 4.4 The average reward variation of V2V-RMM using MDQN and MDDQN, as a function of training episodes.

## 4.5.3 Impact on Network Performance and CAP Quality

### 4.5.3.1 Key Performance Indicators

we compare the performances achieved by V2V-RMM to the ETSI CPM generation rules [37], the dynamics-based RMM [90], and CBR-selective [93] scheme. A CAV utilizing the dynamic-based technique verifies the most recent CPM received from all neighboring CAVs and excludes perceived objects whose positions or speeds exceed static thresholds. We consider the same values defined in [37] for object redundancy for all approaches. An object is redundant if the difference in its absolute speed or position is less than 0.5 m/s and 4 m, respectively.

To show the performances offered by V2V-RMM compared to the above-described works, we defined the following network-related KPIs:

- Object redundancy (OR): indicates the number of times a CAV receives identical information about the same perceived object over the selected time interval.
- Cooperative Perception Awareness (CPA): represents the number of unique objects known to a CAV, given the total number of objects in its coverage. We consider an object to be known by a CAV if it is successfully detected or received via V2V communication.
- CBR: identifies the current utilization percentage of the V2V communication channel. It is determined by assessing the channel in a time interval. Given the duration of one OFDM symbol  $8 \mu\text{s}$  (48 bits per symbol at a data rate of 6 Mbps in the ITS-G5-CCH), the channel is assessed for  $N = 12500$  symbols. Whenever the received signal strength exceeds  $-85 \text{ dBm}$ , the channel is assessed as busy for this symbol. Thus, we define CBR as  $N_{\text{busy}}/N$ .
- CPM delivery (CDR): identifies the probability of correctly receiving a CPM at a given distance  $d$  to the CAV sender. Mathematically, the CDR at CAV  $i$  at within a distance  $d$  is defined by  $\sum_{j=0}^{N_d} \omega_{i,j}(d) / \sum_{j=0}^{N_d} \omega'_{i,j}(d)$ , where  $\omega'_{i,j}(d)$  is the number of CAVs whose Euclidean distances to CAV  $i$  are less than  $d$  when this latter transmits a CPM  $j$ .  $\omega_{i,j}(d)$  is the number of CAVs that successfully receive the CPM  $j$ .  $N_d$  is the number of CPMs sent by CAV  $i$ .

### 4.5.3.2 Simulation Results

We start recording statistics for all network-related KPIs after the training process of the proposal reaches the total number of learning episodes (i.e.,  $N_{\text{episodes}} = 6000$  episodes). Figure 4.5 depicts OR generated by the proposal V2V-RMM using MDDQN and the other approaches of the state of the art. OR is plotted as a function of the distance between the perceived object and the CAV receiving it. We notice that this metric results in high levels at short distances because perceived objects are successfully detected and exchanged by multiple CAVs simultaneously. However, OR reduces with increasing distance because objects become farther

away from the receivers, making their perception more challenging. Figure 4.5 shows that ETSI CPM generation rules generate higher OR levels. This can be explained because CAVs generate and broadcast CPMs without prior intelligence. However, the dynamic-based technique and the CBR-selective scheme have a marginally reduced OR compared to the latter. The proposal shows more efficient OR mitigation at shorter distances of up to 150 m than the other approaches.

The performance achieved by the proposal in mitigating OR shall maintain CAP awareness in the V2V network close to the CPM generation rules, particularly at short distances that are critical for the safety of CAVs. This is shown in Figure 4.6, which depicts CPA level reached by each method as a function of the distance between the detected object and the CAV receiving it. This metric shows that the dynamics-based technique and the CBR-based scheme achieve almost the same CPA as the basic generation rules at distances larger than 100 m; however, their performance degrades by about 5% and 10%, respectively, at distances less than 100 m. On the other hand, the proposed approach significantly improves CPA compared to the other methods over distances less than 200 m, which is critical for the safety of CAVs.

This improvement is achieved by enhancing the reliability of V2V communication, enabling CAVs to receive additional relevant information that is lost or not transmitted by the other methods. We measure this reliability using CDR as depicted in Figure 4.7 CDR is presented as a function of the distance between the transmitter and the receiving CAVs and highlights the increased CPM reception ratio achieved by the proposed solution compared to the other approaches over distances less than 200 m.

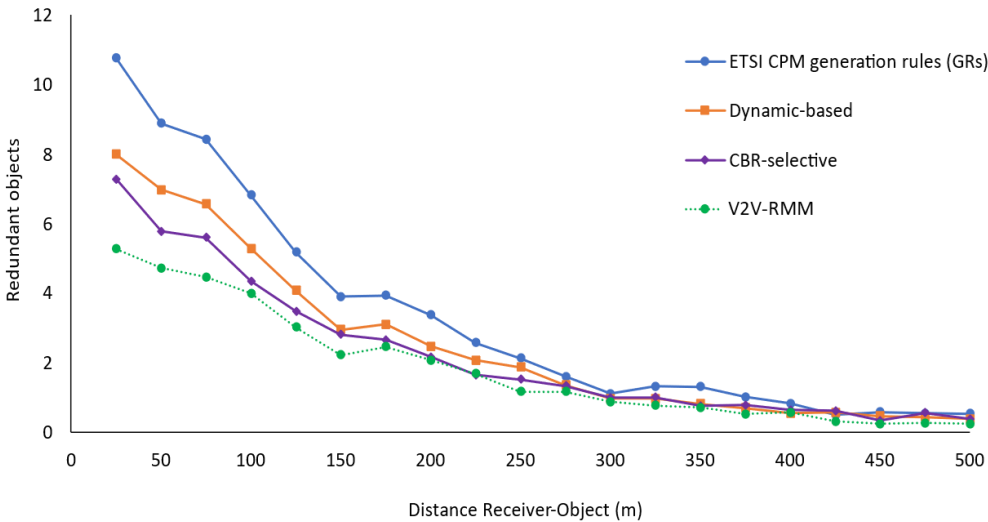


Figure 4.5 Object redundancy as a function of the distance between the perceived object and the CAV receiving it.

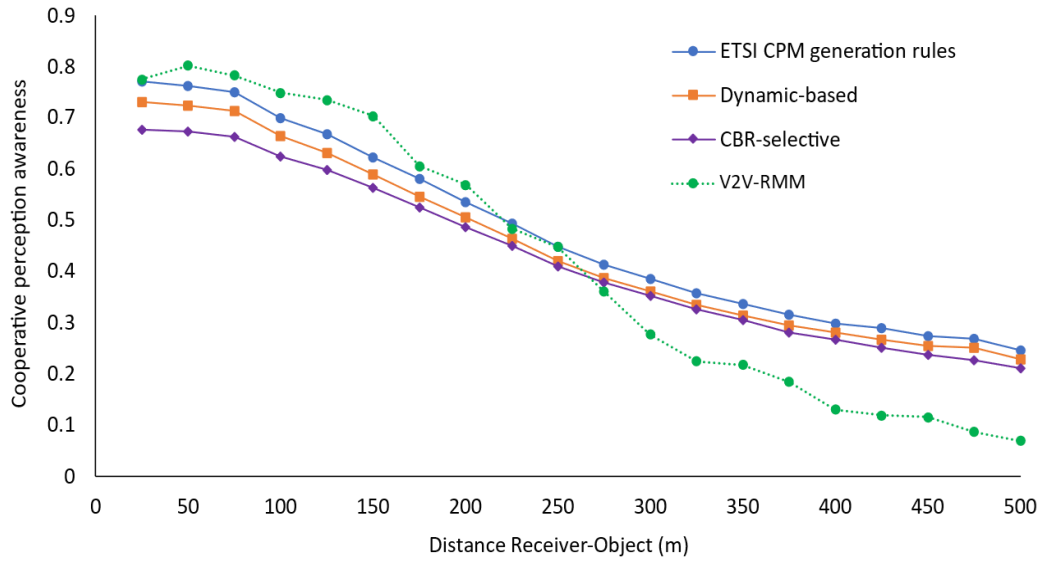


Figure 4.6 CPA as a function of the distance between the perceived object and the CAV receiving it.

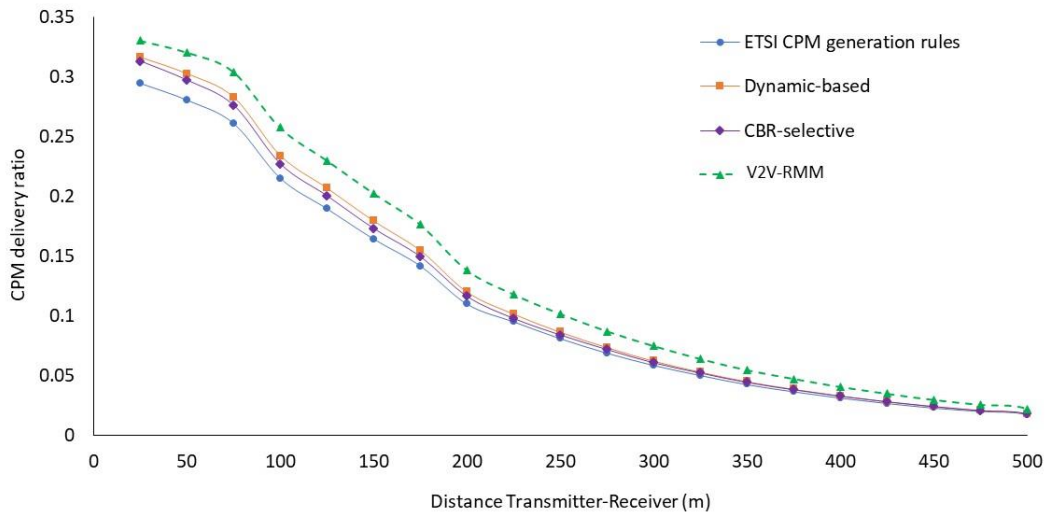


Figure 4.7 CPM delivery ratio as a function of the distance transmitter-receiver

## 4.6 Conclusion

In this chapter, we presented V2V-RMM, a novel and intelligent distributed approach designed to mitigate the transmission of redundant perception information between CAVs while preserving and even enhancing their perception awareness. Our method leverages the power of DRL to learn intelligent strategies for selecting objects to be included in the CPM, considering various critical factors such as distance, object size, viewing angles and occlusions induced by other road users.

The theory behind this method assumes that all objects have the same class priority, regardless of their class. However, it should be noted that objects can actually have varying levels of importance depending on their respective categories. Furthermore, the methods we propose assume a uniform situation for all receivers. This means that, relevance of an object may shift according to the specific driving context of an individual CAV.

The next chapter introduces a context-aware CAP that considers additional perceptual contexts, such as object speed and type, as well as the receiver's situation in its driving context. The development of such approach is vital, particularly in densely congested networks and complex environments, where the transmission of information is considerably limited by the constraints of the communication resources.



# Chapter 5

## Advantage Actor-Critic Learning for Context-Aware V2X Cooperative and Augmented Perception

### 5.1 Introduction

Ensuring CAP quality largely depends on the efficiency of information exchange within the network. In the previous chapter, we presented an intelligent RMM, based on MDRL, aimed at mitigating the transmission of redundant perception information between CAVs. This method considers various perception factors, including distance, object size and road occlusions, in its basic modeling in order to adapt to the dynamic driving environment. The primary objective was to the network reliability and, consequently, the effectiveness of information sent by CAP.

However, as the challenges of modern traffic scenarios continue to evolve, particularly in densely congested networks and complex environments, where RSUs, equipped with advanced on-board sensors, generate and exchange large amounts of perception data with CAVs, information transmission faces significant constraints due to the limitations of communication resources. In this intricate landscape, further refinement of filtering criteria becomes crucial to bolster the quality of CAP under these circumstances.

One promising area for improvement is to adopt more context-aware filtering strategies that consider the relative importance of perceived objects according to their specific categories. For instance, VRU objects generally have the highest priority due to their vulnerability and the significant safety implications associated with them. Similarly, disconnected objects may be assigned a higher priority than connected objects and obstacles due to their dynamic behavior and lack of information-sharing capabilities. Furthermore, the assumption of a uniform context situation for all receivers is no longer adequate in driving environments with increased complexity. This implies that the relevance of information is inherently context-aware and may fluctuate overtime, depending on the unique driving context of each individual receiving CAV.

This chapter introduces MDRL-CR, a novel multi-agent system designed to improve the quality of CAP for CAVs and RSUs. At its core, MDRL-CR utilizes DRL architectures that

employ DNNs to learn relevance-focused filtering and context-aware prioritization policies in tandem. These policies enable CAVs and RSUs to filter and prioritize perceived objects for dissemination through CPMs, based on their specific characteristics and the context of the driving environment each potential receiver. Furthermore, MDRL-CR facilitates the exchange of information across various communication channels, namely MCO [10] and [105], allowing to benefit from excluded information that may be beneficial for further concerns. The proposed system ensures the transmission of relevant and safety-critical perception information over CCH, while directing the rest of less-critical information to SCH2. For the exchange of other types of information, such as those for training purposes, MDRL-CR uses SCH3 for communication.

## 5.2 Related Works

The exchange of CPMs between CAVs and RSUs has emerged as a pivotal solution in overcoming the limitations posed by onboard sensors in vehicular networks, thereby enhancing safety and the overall driving experience. Research has shown that as vehicle mobility and market penetration rates increase, the volume of perception data exchanged between CAVs and RSUs correspondingly rises within the V2X network. While beneficial, this increase can also lead to higher energy consumption, potential network overloads, and degraded communication quality due to excessive and irrelevant data exchanges. This, in turn, could increase data loss and latency, potentially undermining the effectiveness of CAP systems. In addition to the methods mentioned in the previous chapter, several further approaches have been proposed to overcome these challenges. These methods can mainly be grouped into distance-based, DL-based, and RL-based categories.

### 5.2.1 Distance-Based Methods

These methods prioritize data based on distance metrics, considering the position relative to the onboard sensors of the transmitting CAVs to measure relevance of information. For instance, the work proposed in [106] reveals that most existing CAP methods fail to filter out large amounts of less relevant data, thus imposing significant network and computation load, overwhelming the driver, and hindering performance. To address this issue, the authors proposed an Augmented Informative Cooperative Perception (AICP) system that optimizes the relevance of exchanged information between CAVs to improve the perception view. The relevance of information is formulated as a maximization problem at CAVs, employing a fast filtering algorithm based on Mahalanobis distance to select crucial information at the application level, effectively filtering out less relevant packets with minimal delay. Another related work is Cooperverse [107], uses a Dynamic Feature Sharing (DFS) to address the limited communication bandwidth challenges by identifying relevant feature data for sharing through Random-K priorities based on Manhattan distance. A recent study [108] suggested merging different RMMs to manage the generation and transmission of CPMs more efficiently. For example, Look-Ahead [90] generates larger message size by grouping objects into a smaller

number of CPMs. Still, this strategy might increase redundancy due to more frequent object transmissions compared to the standard CPM generation rules. Conversely, the ETSI redundancy mitigation strategies reduce the amount of redundancy transmitted, but they might also increase the generation of small CPMs. This work also emphasized the potential of RL-based methods for optimal outcomes.

## 5.2.2 DL-Based Methods

DL-based methods focus on learning to select CPM information by considering relevant spatial areas. These methods ensure that only relevant information within appropriate communication regions is transmitted, avoiding channel overload. For instance, the authors in [109] leveraged bounding box matching and strategic selection methods to select relevant features included within bounding boxes for transmission. Additionally, the work in [110] introduced a unified bandwidth-efficient and multi-resolution-based CAP, namely UMC. This system encompasses a trainable region-wise communication selection method, leveraging entropy to distinguish informative regions and select suitable regions for transmission based on resolution levels. Another work, GevBEV [111], identifies regions requiring additional information for the transmitting CAV by utilizing evidential Bird's Eye View (BEV) maps [112]. This method employs evidential DL, incorporating Dirichlet evidence [113], to quantify point-based classification uncertainty within BEV maps, aiding in pinpointing the most relevant regions for communication.

## 5.2.3 RL-Based Methods

This category of methods uses RL to select information for transmission. The authors in [114] proposed an online learning-based algorithm known as the Adaptive Volatile Upper Confidence Bound (AVUCB) to schedule the most advantageous receiving CAV, offering an enhanced view while minimizing total power consumption and adhering to communication bandwidth and detection accuracy constraints. The sensor-sharing scheduling problem in this work is formulated as a variant of the Multi-Armed Bandit (MAB) problem, a classical model used in decision theory and RL to describe a scenario where a fixed limited set of resources must be allocated between competing alternatives in a way that maximizes their expected gain under conditions of uncertainty, factoring in perception performance, time-varying wireless channels, and power consumption [115]. However, the AVUCB algorithm does not account for the high mobility of CAVs. To address this mobility challenge, the authors in [116] introduced the Mobility-Aware Sensor Scheduling (MASS) algorithm. MASS, an extension of the AVUCB, formulated CAP scheduling as a Restless-MAB (RMAB) problem. At its core, the modeled rewards continually evolve, requiring each CAV to continuously learn about its surroundings while selecting actions that empirically maximize rewards. When applied to a practical SUMO trace, the MASS algorithm demonstrated the ability to enhance overall perception gain without incurring additional costs associated with frequent meta-information exchanges.

Selecting relevant information to share is vital to improve the quality of CAP. Methods that consider both the state of receiving CAVs and information selection can offer a more balanced approach to optimizing precision and bandwidth in congested and resource-constrained networks. The authors in [117] introduced Where2comm, a novel spatial-confidence-aware CAP system that leverages spatial confidence maps generated for each receiving CAV to identify regions that include relevant features to share via CPMs. A graph describing the communication network is used to determine the target receiving CAVs to effectively minimize unnecessary bandwidth usage, as showcased by the evaluation results. However, a notable gap exists in the consideration of further potential factors that could enhance network reliability and the relevance of the shared information [118] and [119]. These factors could include the prioritization of information in the CPM, tailored to specific conditions encountered by receiving CAVs, under the constraints of network congestion and increased vehicular density. In addition, as the number of non-V2X equipped objects increases, the class and type of objects become crucial in influencing cooperative perception. Safety-critical objects like pedestrians and bicycles should be given higher priority for inclusion in the CPM, in contrast to less critical objects such as road signs, trees, and buildings, which could potentially be sourced from High Definition (HD) Maps [120]. In light of these gaps, it is also important to consider the information that conventional perception methods exclude. While some data might be irrelevant for immediate vehicular concerns, it could hold value for other methods such as lost information recovery, fusion, and security mechanisms.

A strategic approach that simultaneously addresses these challenges, ensuring all CAVs maintain heightened perception awareness, would significantly refine the data-sharing process, thereby enhancing the reliability and quality of CAP systems. The next section unveils our innovative method, MDRL-CR, which is dedicated to developing context-aware and relevance-based CAP strategies for both CAVs and RSUs. This approach simultaneously considers the selection of target receiving CAVs and the relevance of the information to exchange with further related factors, such as object speed and class, aiming to optimize latency and bandwidth usage.

## **5.3 Architecture and Modeling**

This section introduces the architecture and modeling of MDRL-CR.

### **5.3.1 System Components and Communication**

The network architecture of the proposed system, set within a heterogeneous environment of connected and autonomous driving, is depicted in Figure 5.1. This architecture prioritizes perception, processing, and communication, incorporating multiple vehicles of varying dimensions, classes, types, and driving models. The environment also includes road obstacles such as buildings and trees, alongside VRUs like pedestrians, cyclists, and motorcyclists, who navigate with heightened risks and necessitate enhanced safety measures. Vehicles are equipped with GPS and GNSS for precise positioning and feature onboard sensors, including radars, lidars, and cameras, providing a 360° FoV to detect road objects. CAVs utilize V2X

communication to interact with each other and with RSUs, which are also equipped with sensors and relay additional traffic data to CAVs within their communication range. Both CAVs and RSUs can connect with Edge computing, serving as a central node for additional computational and communication resources. To facilitate these exchanges, distinct communication scenarios are outlined as follows: V2V, Vehicle-to-RSU (V2R), RSU-to-Vehicle (R2V), Vehicle-to-Edge (V2E), and RSU-to-Edge (R2E). V2R and R2V facilitate exchanges between CAVs and RSUs, while V2E and R2E involve interactions with Edge computing. The proposed system operates on multiple channels using MCO. It ensures that V2V and R2V interactions occur over the CCH, while separate SCHs are used by CAVs and RSUs for communications with the Edge computing node, covering the V2R, V2E, and R2E scenarios.

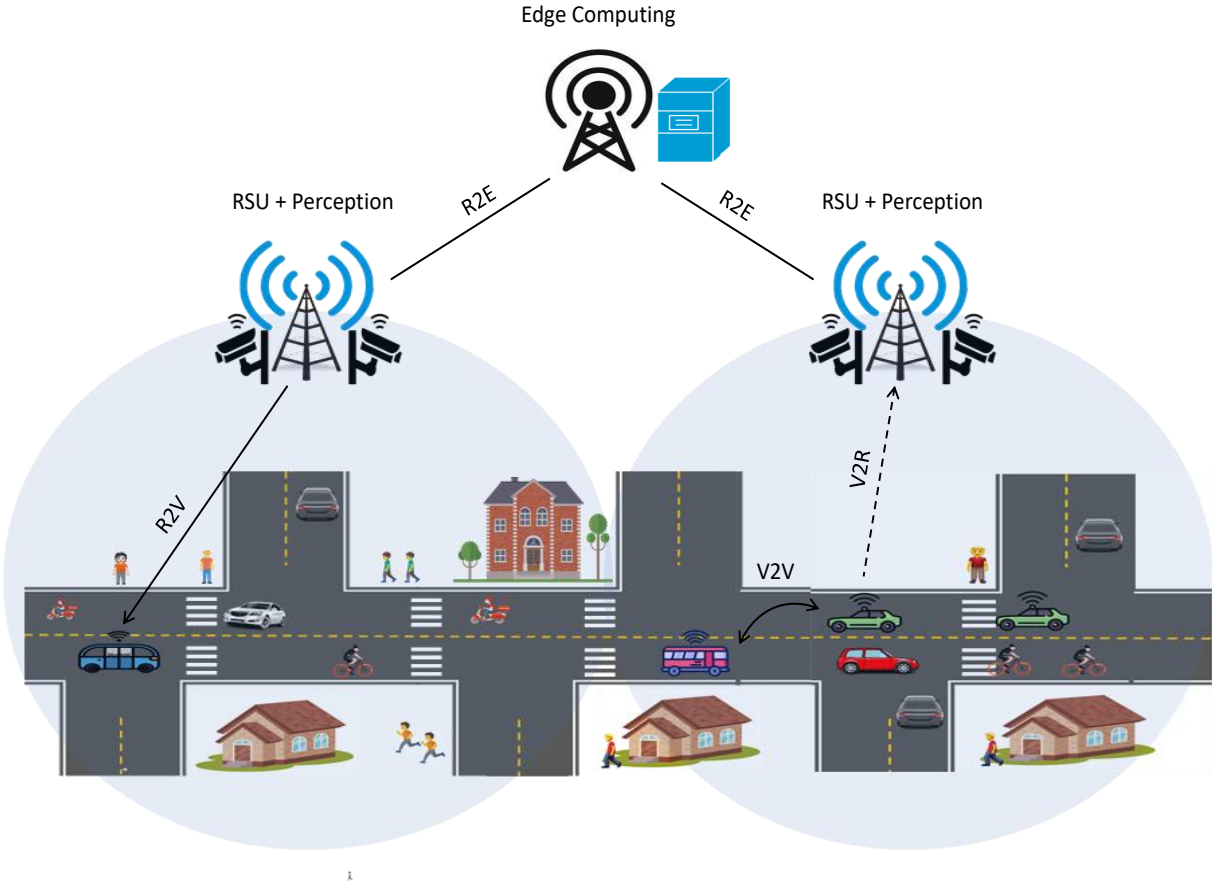


Figure 5.1 Network architecture within a heterogeneous environment of connected and autonomous driving.

### 5.3.2 Functional Architecture

Based on the aforementioned definitions and communication scenarios, each CAV and RSU employs two main services: LPS and CAS as depicted in Figure 5.2. LPS processes the CAV or RSU kinematic information and raw sensory data from its onboard sensors to detect and classify road objects. The perception at this stage is enhanced by HD map, which improves the

data fusion process and helps identify static objects like road signs, buildings, and trees. Meanwhile; CAS gathers kinematic information from GPS/GNSS, broadcasting status updates via CAMs at defined intervals  $t_{av}$  for CAVs, as defined in the previous chapter, and  $t_{ar}$  for RSUs, using CCH. The status information of a designated CAV of index  $i$  is the same as defined previously, however it also capture the speed  $sp_i$  of the CAV, denoted by  $h_i(t_{av}) = \{x_i(t_{av}), y_i(t_{av}), sp_i(t_{av}), l_i, w_i\}$ . For a given RSU  $j$ , only its position is considered, indicated by  $h_j(t_{ar}) = \{x_j(t_{ar}), y_j(t_{ar})\}$ . Furthermore, we consider also a CPS as additional core service to enable each CAV and RSU to broadcast and receive information about detected objects via CPMs at intervals of  $t_{pv}$  for CAVs and  $t_{pr}$  for RSUs, using both CCH and SCH2. Within this framework, the CPS comprises two key modules, as illustrated Figure 5.2:

- Information Management and Provider (IMP): This module fuses data received from various sources, including the LPS, HD map, neighboring CAVs, and the associated RSU. Its main role is to produce a comprehensive list of detected objects and provide a description of the driving environment as outputs.
- MDRL-CR: This module filters and prioritizes perception data received by IMP based on the description of the environment, for inclusion in the CPM. The CPM is then transmitted through the CCH and SCH2. Additionally, this model facilitates the exchanges of training data with the Edge computing node through SCH3.

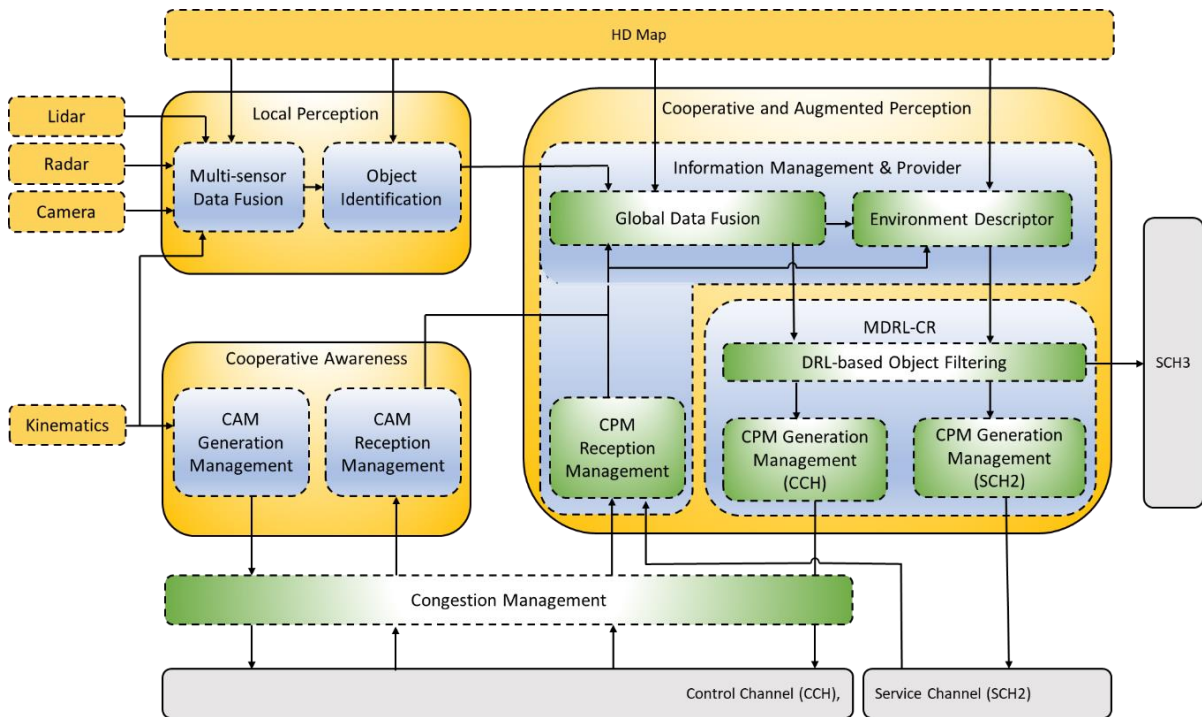


Figure 5.2 Architecture of MDRL-CR

### 5.3.3 Context-Aware Modelling

Our analysis focuses on designated time intervals  $t$ , distinguished as  $t_{pv}$  for CAVs and  $t_{pr}$  for RSUs. The list of detected objects given by the IMP of a given CAV or RSU of index  $i$  at  $t$  can be denoted by  $T_i(t) = \{o_{i,1}(t), o_{i,2}(t), \dots, o_{i,k}(t)\}$ . Each object, denoted as  $o_{i,j}(t)$ , within this list is characterized by its coordinates  $x_j(t)$  and  $y_j(t)$ , speed  $sp_j(t)$ , class  $cl_j$ , and dimension  $l_j$  and  $w_j$ . Given the potential for high traffic density, the sheer volume of detected objects presents a challenge. Directly broadcasting the complete list via CPMs risks overwhelming the communication channel. To address this, we introduce a sophisticated set of criteria designed to efficiently filter and prioritize the perception data for broadcast, ensuring the dissemination process is both intelligent and effective.

#### 5.3.3.1 Object Categorization and Prioritization

The proposed system categorizes detected objects by CAVs and RSUs into four primary categories to improve the selection process:

- **VRU:** This category includes objects of type VRU, such as pedestrians, cyclists, and also motorcyclists, emergency vehicles, road workers, and animals. Due to their lack of V2X capabilities, these objects cannot actively communicate with CAVs and RSUs and require increased safety measures.
- **Obstacles:** This group encompasses objects that are typically stationary and can affect navigation paths, such as road signs, buildings, and trees.
- **Connected:** Represents objects equipped with V2X technology, such as passenger cars, buses, and trucks.
- **Disconnected:** This groups objects that are not equipped with V2X, such as non-connected vehicles.

Given these classifications, the prioritization of objects for communication and processing is dynamically adjusted. VRU objects are accorded the highest priority reflecting their increased vulnerability and the critical safety implications of failing to detect and respond to them promptly. Disconnected objects are also prioritized over Connected objects and stationary Obstacles due to their unpredictable behavior and absence of communication capability.

To that end, the priority  $P(o)$  assigned to object by  $o$  is calculated as follows:

$$P(o) = \begin{cases} \alpha_1, & o \in \text{VRU} \\ \alpha_2, & o \in \text{Disconnected} \\ \alpha_3, & o \in \text{Obstacles} \\ \alpha_4, & o \in \text{Connected} \end{cases}, \quad (5.1)$$

With

$$C1: \quad 1 \geq \alpha_1 > \alpha_2 > \alpha_3 > \alpha_4 \geq 0 \quad (5.2)$$

Constraint C1 ensures that categories have different values with certain order in  $[0,1]$ .

### 5.3.3.2 Driving Context Evaluation

Each CAV navigates through a unique driving environment shaped by an array of neighboring road elements. These elements exhibit a wide spectrum of characteristics and dynamics, from stationary obstacles like building and trees to dynamic elements such as vehicles and VRUs. To systematically evaluate the contextual challenges faced by a transmitting CAV  $i$  within its Safety Coverage  $SC_i$  at time  $t$ , we propose the concept of a Driving Context Score (DCS), denoted as:

$$DCS_i(t) = 1 - \frac{\sum_{j \in SC_i(t)} \left( \left( 1 - \frac{d_{i,j}(t)}{SC_i(t)} \right) * \frac{1}{\sqrt{sp_j(t)}} \right)^{1-P(j)}}{n} \quad (5.3)$$

The driving context score, which ranges from 0 to 1, incorporates multiple factors, such as the distance  $d_{i,j}$  to each adjacent road element  $j$  and its dynamics within  $SC_i$ . The objective is to highlight the impact of road elements that are moving towards the CAV at high speeds, thereby increasing the complexity of its driving context. Furthermore, this driving score considers the density  $n$  and categories of road elements  $P(j)$  that exist currently in  $SC_i$  at time  $t$  in the modeling, giving greater attention to congested environments with higher densities and increased number of disconnected road elements and VRUs. For example, a CAV operating in a densely urban area, teeming with pedestrians, cyclists, and a mix of connected and disconnected vehicles, will register a higher DCS, signaling the elevated complexity and potential risk factors of its current context. Conversely, a CAV traversing a sparsely populated rural road, with fewer dynamic obstacles and less frequent interactions with other road users, will have a lower DCS, reflecting a less challenging driving scenario.

This comprehensive evaluation aids in quantifying the contextual challenges currently faced by each potential receiving CAV during the CPM transmission process, facilitating more informed decision-making, enabling to prioritize actions that mitigate risks by augmenting awareness in high-density and high-complexity environments.

### 5.3.3.3 Considerations of Distance, Speed, and Road Obstructions

The relevance of objects within a receiver's detection range is influenced by their relative position and occlusion as and velocity. Specifically, objects that are distant from the receiver and those moving at speeds are typically considered more relevant, as detecting them with onboard sensors becomes increasingly complex. Additionally, road obstructions can obscure the receiver's LoS, impairing its ability to detect specific objects. To address these considerations simultaneously, we introduce  $F_{o,r}$ , a measure that assesses the capability of a



receiver  $r$  to observe an object  $o$  within its sensing range at a given time  $t$ . This metric is already defined in Equation (4.8) and Equation (4.9), however we consider the effect of the speed factor as follows:

$$F_{o,r}(t) = \begin{cases} 0, & d_{o,r} > m_r \\ \frac{f_{o,r}(t) * g_{r,o}(t)}{\sqrt{sp_o(t)}}, & otherwise \end{cases} \quad (5.4)$$

### 5.3.3.4 All in One Optimization Problem Formulation

The primary goal of the MDRL-CR is to refine the inclusion criteria for objects in CPM, focusing on the joint factors of object relevance and state of receivers for CAVs and RSUs. To achieve this, we formulate the following optimization problem:

$$\begin{aligned} & \text{maximize} \quad \sum_{r \in C_i(t)} DCS_r(t) * \sum_{o \in M_i(t)} (1 - F_{o,r}(t))^{1-P(o)} \\ & \text{Subject to:} \end{aligned} \quad (5.5)$$

$$C2: \quad M_i(t) \subseteq T_i(t)$$

$$C3: \quad t_{av}, t_{ar} < t < 2 * (t_{av}, t_{ar})$$

Here, the objective of each transmitting CAV or RSU of index  $i$  at the corresponding CP time  $t$  is to filter a set of detected objects  $M_i(t)$  from  $T_i(t)$  to include in the CPM. This filtering focuses on considering simultaneously the different factors defined above.  $C_i(t)$  represents the set of possible receivers in the sender's communication range. Constraint C3 ensures that CAVs and RSUs exchange status updates to process new information using their IMP before starting the filtering process.

## 5.4 Learning Algorithm for Context-Aware CAP

This section introduces the system modeling and learning algorithm of MDRL-CR.

### 5.4.1 Modelling of the Multi-Agent RL Environment

In our proposed system, the vehicular environment is modeled as a multi-agent RL framework where CAVs and RSUs act as agents. The primary role of these agents is to address the combinatorial challenges posed by the multi-criteria optimization of Equation (5.5). At each

CPM generation time  $t$ , agents perform three key functions: build an environmental state, select an action based on this state, and receive a reward corresponding to the action's effectiveness.

The environment state of the  $i$ -th agent at time  $t$ , denoted as  $s_i(t)$ , is constructed based on the information provided by its IMP module, as follows:

$$s_i(t) = \{E_i(t), T_i(t)\}, \quad (5.6)$$

where

$$E_i(t) = \left\{ \left( d_{i,j}(t), \alpha_{i,j}(t), l_j, w_j, sp_i(t), DCS_j(t) \right) \right\}, \quad (5.7)$$

denotes the set of all agents within its communication coverage. Each agent of index  $j$  is characterized by the distance  $d_{i,j}$  and viewing angle  $\alpha_{i,j}$  with respect to the agent  $i$ . We consider also its dimension, speed and driving context score at the same time instance. This provides a dynamic and coordinate-independent representation crucial for agents as they move and store decision experiences to enhance future behavior. The state also includes all identified objects given by IMP, delineated by attributes such as dimensions, speed, and classes, to ensure decision-making is invariant to fluctuating positional data. To that end, the FoV of each agent is represented as a matrix of  $I * J$  rectangular cells, where each cell is capable of holding a single object, as follows.

$$Cells_i(t) = \begin{bmatrix} C_{1,1}^1(t) & C_{1,2}^2(t) & \cdots & C_{1,J}^J(t) \\ C_{2,1}^{J+1}(t) & C_{2,2}^{J+2}(t) & \cdots & C_{2,J}^{2*J}(t) \\ \vdots & \vdots & \ddots & \vdots \\ C_{I,1}^{(I-1)*J+1}(t) & C_{I,2}^{(I-1)*J+2}(t) & \cdots & C_{I,J}^{I*J}(t) \end{bmatrix} \quad (5.8)$$

To accomplish our objective, the action space is defined as the power set of  $Cells_i(t)$ , encompassing all possible object combinations. An action taken by an agent  $i$  at time  $t$ , designated by  $a_i(t)$ , maps to natural numbers in the range  $[0, 2^{I*J} - 1]$ , which is then translated into a binary string  $(b_0 b_1 \dots b_{I*J-1})_2$  of length  $I * J$ . The inclusion of objects in the current CPM is contingent upon the corresponding binary indicators: a '1' denotes inclusion, whereas a '0' signifies exclusion.

Lastly, the reward for the agent  $i$ , denoted as  $r_i(t)$ , is calculated based on the selected action's outcome set  $M_i(t)$  at time  $t$ :

$$r_i(t) = \sum_{r \in C_i(t)} DCS_r(t) * \sum_{o \in M_i(t)} (1 - F_{o,r}(t))^{1-P(o)} \quad (5.9)$$

This function encapsulates the effectiveness of the detected objects within the agent's communication range.

## 5.4.2 Advantage Actor-Critic Learning

The aim of MDRL-CR is to learn agents to optimize their cooperative perception in tandem with the dynamic nature of the driving environment. RL algorithms are mainly categorized into value-based and policy-based. Value-based methods, like the DQN and DDQN [100], seek to formulate the optimal policy by valuing each potential action or state. Their effectiveness, however, is often constrained in complex environments with large state and action spaces due to the increased computational demands [121]. Conversely, policy-based RL algorithms, such as policy gradient [100] gradually fit and evaluate a policy without the need to explore entire spaces. Still, this category tends to generate a large variance when estimating the gradient since updates usually occur per episode, which reduces training efficiency.

To enhance RL-based systems in complex scenarios, an Actor-Critic (AC) algorithm was proposed, combining the strengths of both value-based and policy-based methods [122]. A fundamental challenge with AC algorithms is the generation of high variance in its policy updates. This variance stems from the dependency of the policy gradient on the expected return from a state-action pairs, which can be inherently noisy due to the stochastic nature of the policy and the environment. The critic's value function approximation, intended to reduce this variance, may itself be imprecise, potentially introducing additional noise and increasing the variance instead of mitigating it. Consequently, these noisy and high-variance updates can lead to slower convergence and cause the algorithm to converge to suboptimal policies [123]. To address the issue of training variation, an A2C algorithm was introduced [124]. This algorithm differentiates between the value of a state and the advantage of taking a specific action from that state. The actor selects an action based on its policy, while the critic evaluates this action and returns an advantage score. This feedback allows the actor to adjust its policy more accurately, thereby reducing the noise in policy updates and leading to more stable and consistent learning.

---

**Algorithm 3:** The training algorithm of the proposed systems

---

```
1: Initialization:
2:   |  $N_{ep}C, N_{ep}R$ : number of episodes for CAVs and RSUs, respectively.
3:   |  $N_{steps}C, N_{steps}R$ : number of steps by episode for CAVs and RSUs, respectively.
4: Output:
5:   | Trained model for each CAV and RSU
6: Begin:
7:   | Do in parallel:
8:     | Train (MARL-CAV,  $N_{ep}C, N_{steps}C$ )
9:     | Train (MARL-RSU,  $N_{ep}R, N_{steps}R$ )
10:  | end do
11: end
```

---

---

**Algorithm 4:** Train (sub-system,  $N_{ep}$ ,  $N_{steps}$ )

---

```
1: Initialization:
2:   A critic network  $Q$  with random weights  $\theta^-$  and a buffer of experiences  $\mathcal{D}$  on the
   edge computing device.
3:   An actor network  $\pi$  with random weights  $\theta$  for each agent associated with sub-
   system.
4:   Discount factor  $\gamma$  and learning rate  $\alpha$  in  $[0,1]$ 
5: Begin:
6:    $ep \leftarrow 0$ 
7:   while  $ep < N_{ep}$  do
8:      $t \leftarrow 0$ 
9:     while  $t < N_{steps}$ , for each agent associated with sub-system do
10:      Build  $s(t)$ , select  $a(t)$ , and get  $r(t)$ 
11:      Store  $e(t) = ((s(t), a(t), r(t), s(t + 1)))$  into  $\mathcal{D}$ 
12:       $t \leftarrow t + 1$ 
13:     End while
14:     Sample  $(s, a, r, s + 1) \sim U(\mathcal{D})$  of size  $n_e$ 
15:     Compute the critic loss:
           
$$Loss(\theta^-) = \left(\frac{1}{n_e}\right) \sum_{t=0}^{n_e-1} \left(r(t) + \gamma Q_{\theta^-}(s(t+1)) - Q_{\theta^-}(s(t), a(t))\right)^2$$

16:     Update the weights of the critic model:
           
$$\theta^- = \theta^- + \alpha * Loss(\theta^-)$$

17:     Assess how good the chosen action  $a$  is in the state  $s$ :
           
$$A(s(t), a(t)) = r(t) + \gamma Q_{\theta^-}(s(t+1)) - Q_{\theta}(s(t), a(t))$$

18:     Compute the gradient of the policy model:
           
$$\nabla_{\theta} J(\theta) = \sum_{t=0}^{|\tau_i|-1} \nabla_{\theta} \log \pi_{\theta}(a(t)|s(t)) A(s(t), a(t))$$

19:     Update the weights of the actor model:
           
$$\theta = \theta + \alpha * \nabla_{\theta} J(\theta)$$

20:      $ep \leftarrow ep + 1$ 
21:   end while
22: end
```

---

Transitioning to MDRL setting, the Centralized Training with Decentralized Execution (CTDE) paradigm has been utilized to construct a multi-agent AC environment known as Centralized Critic for Decentralized Actors (CCDA) [125]. Inspired by this, we adopt the same framework for the development of our proposed MDRL-CR system, detailed in Algorithm 3. It is essential to recognize the distinct perception capabilities of CAVs and RSUs. RSUs, for

instance, are typically equipped with sensors that have a greater range and superior detection capabilities compared to CAVs. Therefore, we divide the primary challenge addressed by MDRL-CR into two sub-problems, specifically tailored for CAVs and RSUs, separately. For the CAV-related optimization problem, we employ MDRL-CAV, while MDRL-RSU is utilized for RSUs. The training process for both sub-systems is essentially identical, however, with different parameters

Algorithm 4 delves into the core of the training method. The Edge computing node starts by initializing a critic network  $Q$  using random parameters  $\theta^-$ , and setting up an experience buffer  $\mathcal{D}$  for each sub-system. Subsequently, each agent, either a CAV or RSU, initializes a policy network  $\pi$  with random parameters  $\theta$ . The learning process is conducted over  $N_{episodes}$  episodes for each sub-system. During each episode, agents go through a series of steps  $N_{steps}$  times. Initially, an agent builds the state  $s$  of its environment using data provided by its IMP. It then selects an action  $a$ , generates objects, and broadcasts an optimized CPM through CCH. Objects that are omitted from this CPM are transmitted via a new CPM over SCH2. Following this, the agent then receives a reward  $r$ , and sends an experience tuple  $e = (s, a, r, s + 1)$  to the edge computing node via SCH3. These experiences are stored in the corresponding buffer. At the end of each episode, the critic model of each sub-system on the edge computing node, and the associated agents adjust their weights to improve their behaviors. The adjustment of the critic weights is done by computing the Temporal Difference (TD) error (Lines 14-16). This error quantifies the deviation between the anticipated and outcomes from a uniformly sampled minibatch of experiences  $(s, a, r, s + 1) \sim U(\mathcal{D})$ , of size  $n_e$ , drawn from the buffer of experiences  $\mathcal{D}$ . Agents, in contrast, update their policies by computing the gradient using the policy gradient method [126].

## 5.5 Simulation Results and Evaluations

The primary objective of MDRL-CR revolves around the enhancement of CAP effectiveness in CAVs and RSUs through the MDRL-CAV and MDRL-RSU sub-systems

### 5.5.1 Scenario and Communication Parameters

We define multiple RSUs and a high number of vehicles with diverse dimensions, types (both connected and disconnected), and classes, as well as road obstacles on a select map from OpenStreMap. Table 5.1 provides a summary of the parameters used to evaluate MDRL-CR.

In terms of perception, the connected vehicles, or CAVs in our simulation, are equipped with 360° radar, lidar, and camera sensors, having a sensing range of up to 100 m. In contrast, the RSUs are designed with a broader sensing horizon of up to 300 m. The FoV for both CAVs and RSUs is systematically divided into rectangular cells, sized on average vehicle dimensions. From a communications side, CAVs and RSUs are configured to exchange CAMs and CPMs at intervals of 100 ms and 150 ms, respectively, based on the ETSI ITS-G5 standard. CAMs and CPMs optimized by our model are exchanged through CCH between CAVs and RSUs via

V2V and R2V communication scenarios. Additional interactions occur through SCH2 and SCH3, as previously described.

<b>Parameter</b>	<b>Value</b>
Communication Profile	ITS-G5
CAM Generation Time	$t_{av} = t_{ar} = 100ms$
CPM Generation Time	$t_{pv} = t_{pr} = 150ms$
Primary Channel for V2V and R2V: Used for relevant data, including CAMs and CPMs produced by MDRL-CR	CCH
Secondary Channel: Conveys less crucial data, especially for objects not included in the CPM.	SCH2
Tertiary Channel: Used for V2R, V2E, and R2E exchanges	SCH3
Transmission power	23 dBm
Data rate	6 Mbps
Sensors	Radar,Lidar, Camera
View	360°
range (CAVs, RSUs)	(100m, 300m)
Average vehicle size (length, width)	(4m, 2m)
Object priorities ( $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ )	(1, 0.75, 0.5, 0.25)
Safety coverage range	100 m
Simulation time	70000 time slots
Number of episodes	5000
Number of steps by a single episode	10

Table 5.1 Parameters used for simulation of MDRL-CR

## 5.5.2 Evaluation of The Training Phase: Reward and Loss Variation

Using the PyTorch library, we developed the MDRL-CR system and evaluated its over 70000 time slots, each corresponding to a single CPM generation time. We divided the evaluation into two phases: training and exploitation. During the training phase, MDRL-CR learns optimal policies for CAVs and RSUs over 5000 episodes, with each episode encompassing 10 CPM generation times. The selected hyperparameters for this phase for this phase included a learning rate of 0.001, a minibatch size of 64, a discount factor of 0.99, and an experience buffer size of 1000000.

We start by studying the performance of the training phase of MDRL-CR. Figure 5.3 and Figure 5.4 illustrate the variation in average reward and training loss as a function of learning episodes, respectively. The reward at a specific episode, as shown in Figure 5.3 represents the

average value of the CPMs generated from the actions taken by CAVs and RSUs, Figure 5.4 also indicates the average maximum CPM values that can be attained.

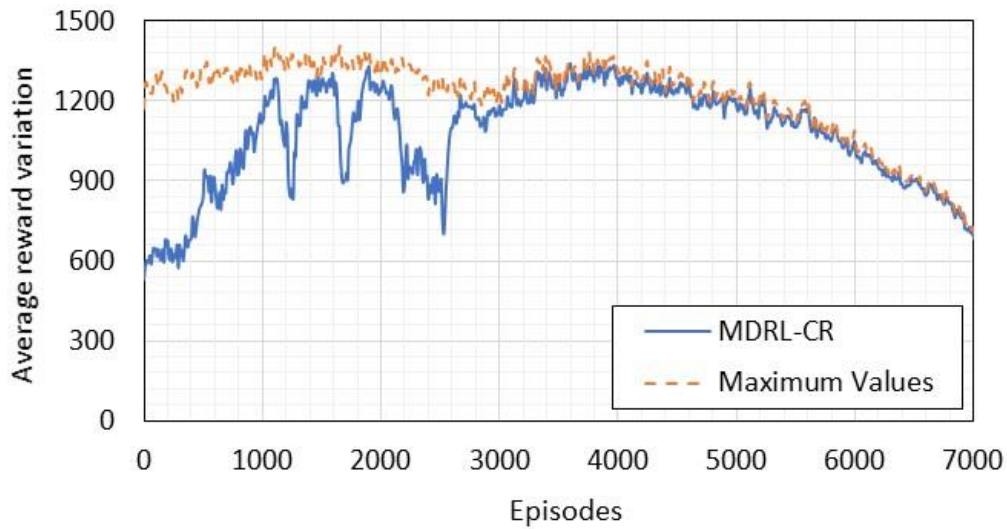


Figure 5.3 Average reward variation of MDRL-CR

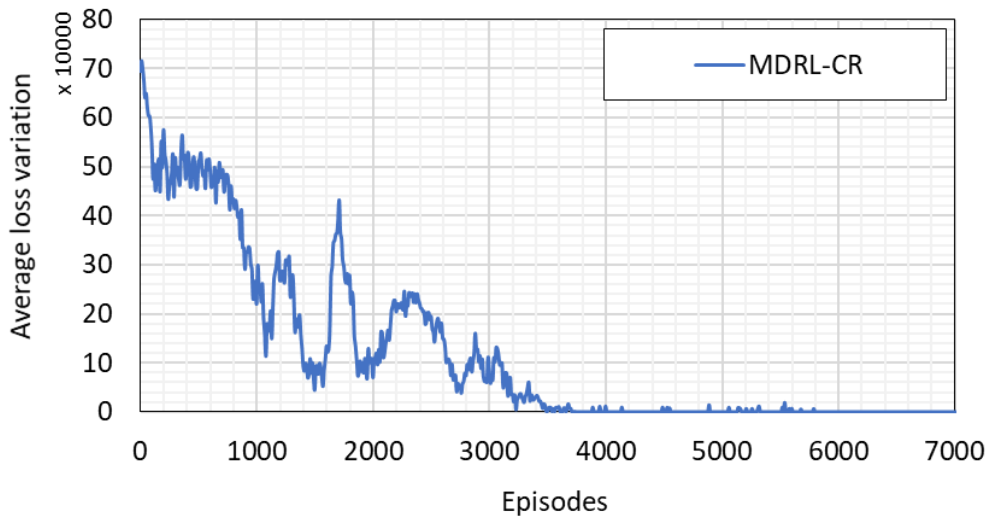


Figure 5.4 Average loss variation of MDRL-CR

During the training phase, the parameters of the actor network of each CAV and RSU and the critic networks on the edge computing device are updated at the end of each episode to minimize their losses. The actor networks are adjusted in a direction that increases the probability of selecting cells that lead to higher rewards, while the critic networks refine their parameters to better estimate the value functions of the current associated policies. The variation of the training loss depicted in Figure 5.4 reflects how quickly MDRL-CR learns optimal policies for CAVs and RSUs, simultaneously. This value is calculated as an average between the actor's and critic's losses at each episode.

As shown in Figure 5.4, the average reward initially rises from low values and then gradually increases with each episode. Concurrently, the training loss begins at high values with large variations, indicating that the training is always far from convergence. As learning advances, we observe a decrease in loss, an increase in reward, and a reduction in variation, signifying a move toward an optimal convergence of policies. The fluctuations seen between episodes 1000 and 4000 may denote the model's encounters new states or unexpected challenges, such as new traffic conditions, which it overcomes as indicated by the next episodes. Post 4,000 episodes, the training of MDRL-CR appears to converge to optimal policies, demonstrated by near-zero loss values and the achieving of maximum reward levels. Note that a decrease in the reward beyond this point does not necessarily imply a decline in training quality, as maximum achievable values also decrease, while low loss values persist.

### **5.5.3 Evaluation of The Exploitation Phase: Impact on Cooperative Perception and Network Reliability**

This section evaluates the performance of MDRL-CR in enhancing network reliability and thereby the quality and effectiveness of CAP systems. For this purpose, we also consider a comparison with two recent works from the state-of-the-art. The first work proposes three different techniques for scalable cooperative perception: LARM, RMLA, and eRMLA [110]. Briefly, LARM combines Look-Ahead (LA) [90] with Redundancy Mitigation (RM) [127] to select objects, initially using the CPM generation rules and LA, followed by RM to mitigate redundancy, thus reducing unnecessary channel load. Conversely, RMLA prioritizes RM and then applies LA to objects that do not satisfy the generation rules, aiming to transmit as many relevant objects as possible. However, the authors highlight some potential inefficiencies in these techniques. When using RMLA, LA could include redundant objects in the CPM since RM is applied first. In the case of LARM, RM might remove objects that LA has anticipated, because LA precedes RM, potentially resulting in frequent CPMs with a low number of objects and an overloaded communication channel due to the significant size of headers and containers. The eRMLA technique addresses these issues by applying an enhanced LA to all detected objects, even those initially considered redundant, to maximize the inclusion of objects in the CPM. Evaluations show that eRMLA yields superior performance; therefore, we consider it exclusively for comparison with MDRL-CR, adhering to the parameters specified in [108]. The second work considered for comparison is an improved version of V2V-RMM, namely MCORM (A2C). This version uses the A2C algorithm based on CCDA, where CAVs learn policies to exchange efficient CPMs to solve the optimization problem of the previous Chapter 4, denoted in Equation (4.5).



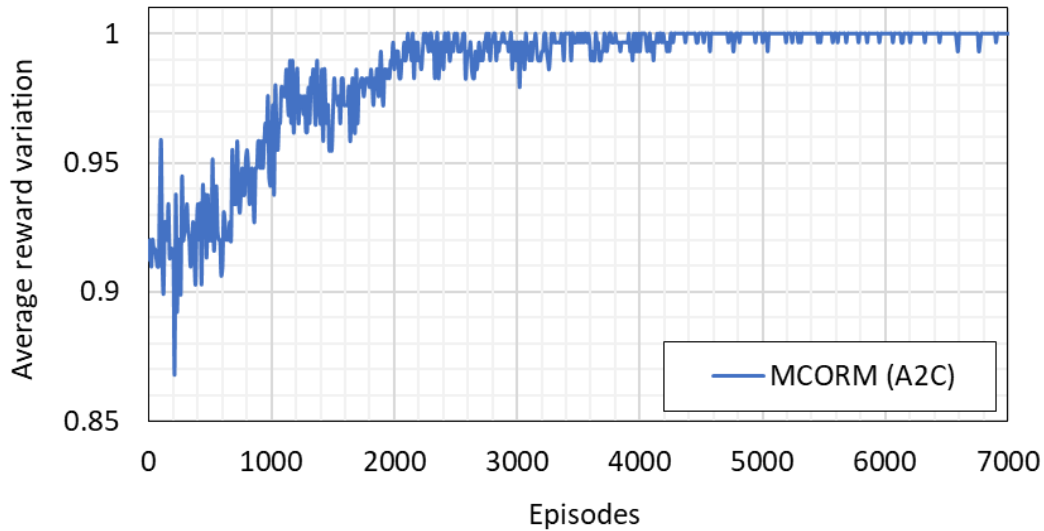


Figure 5.5 Average reward variation of MCORM using A2C algorithm, as a function of training episodes.

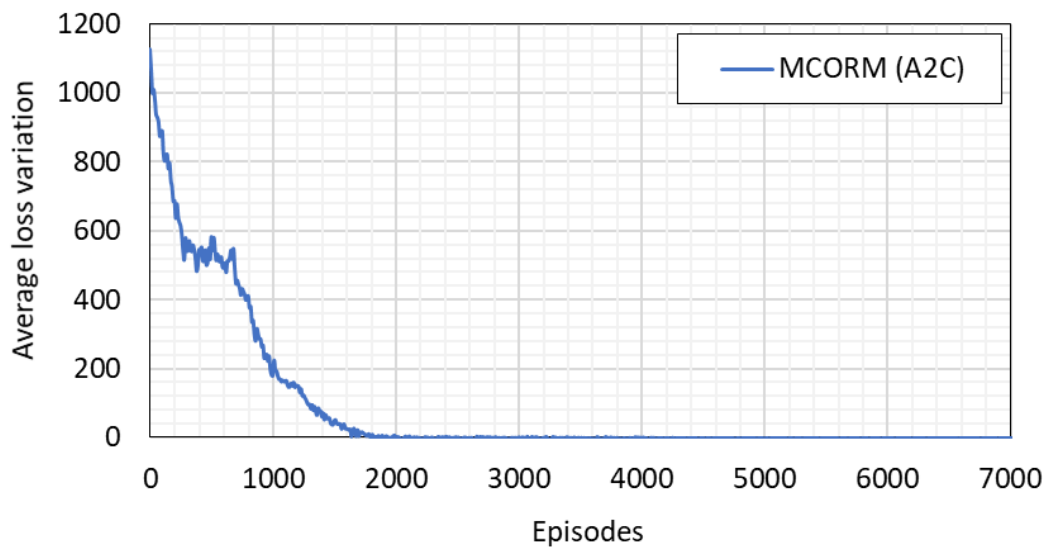


Figure 5.6 Average loss variation of MCORM using A2C algorithm, as a function of training episodes.

We consider the same training and exploitation phases outlined for MDRL-CR, using to the same hyperparameters and assumptions. Figure 5.5 and Figure 5.6 suggest that MCORM's average reward and loss variations likely converge during the training phase, which sets the stage for exploiting the learned policies in our comparative study during the exploitation phase.

The primary objective of MDRL-CR is to learn policies for CAVs and RSUs that focus on relevance-based filtering and context-sensitive prioritization. These policies are used to filter and prioritize perception data for dissemination via CPMs, considering the specific environmental context of each receiving CAV. Figure 5.7 illustrates the average number of VRU and Disconnected objects received by CAVs from neighboring vehicles and RSUs, using eRMLA, MCORM, and the MDRL-CR methods. We observe that in low-critical driving contexts (with a DCS up to 0.4), CAVs receive a relatively low number of VRU and Disconnected objects, which correspond with the lower speed values, as depicted in Figure 5.8, across all methods. This observation aligns with scenarios of sparse vehicular density within the driving safety coverage, which are less likely to contain numerous road elements, including VRUs and Disconnected objects.

However, as DCS increases beyond 0.4, there is a noticeable increase in both the number and speed of these objects, indicating more congested safety coverage zones. In such scenarios, the necessity for receiving CAVs to access more relevant and safety-critical perception data becomes paramount. Figure 5.7 shows that MCORM (A2C) slightly outperforms eRMLA in highly critical driving contexts with DCS greater than 0.8, while the speed difference, as shown in Figure 5.8, is negligible. However, the limitation of MCORM (A2C) lies in its assumption of uniform context scores across the driving environment, which fails to address the varying needs of CAVs in different critical situations. In contrast, the MDRL-CR model shows superior performance in medium to high-critical contexts. It effectively provides receiving CAVs with a greater number of VRUs and Disconnected objects that are often closer and moving at higher speeds, reflecting its adaptability to dynamic driving conditions. These results highlight the effectiveness of MDRL-CR in enhancing cooperative perception by ensuring the delivery of relevant and safety-critical information in challenging driving conditions.

The performance of MDRL-CR in enhancing object perception ratio is evaluated and compared to eRMLA and MCORM (A2C), based on communications between CAVs (V2V) exclusively, and then integrating interactions with roadside units (V2V+R2V). Figure 5.9 depicts this metric as a function of the distance from the receiving CAV, quantifying the ratio of unique objects received by CPMs to the total number of objects within its communication coverage. MDRL-CR distinguishes itself by utilizing MCO across both CCH and SCH2, while eRMLA and MCORM (A2C) operate solely on CCH. All methods achieve a high object perception ratio at smaller distances using V2V; however, performance degrades at larger distances due to the limitations of CAVs' onboard sensors and the effects of propagation and interference. Nevertheless, the integration of R2V significantly improves the performance of all methods, maintaining a high object perception ratio even at a distance of 500 m, highlighting the significant role of RSUs in augmenting perception awareness.

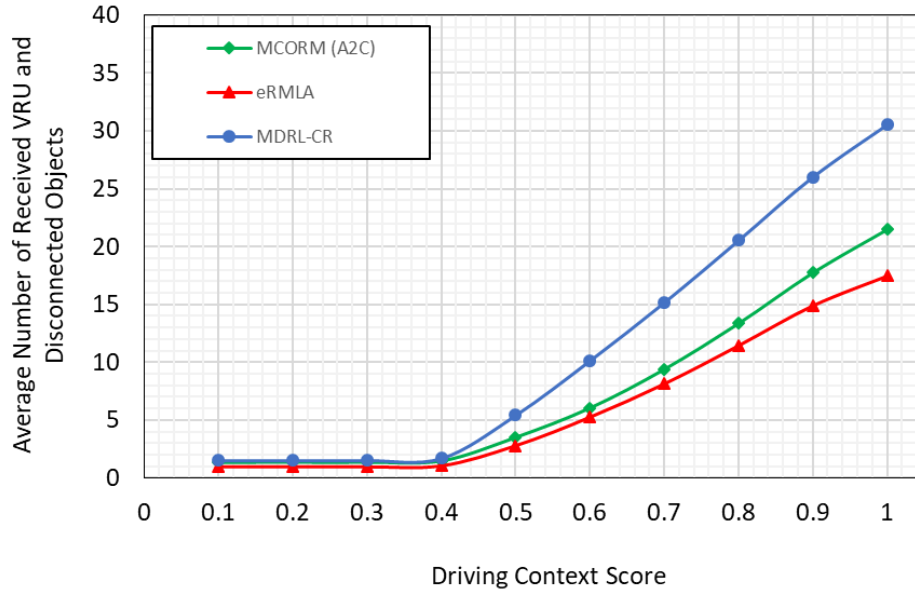


Figure 5.7 Variation of the average number of VRU and Disconnected objected received by CAVs as a function of their DCS.

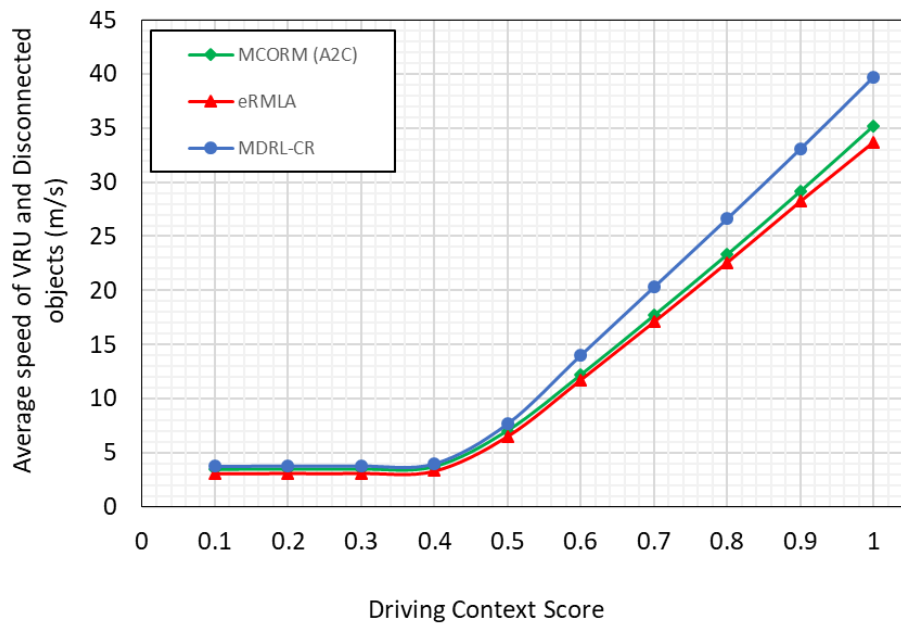


Figure 5.8 Variation of the average speed of VRU and Disconnected objected received by CAVs as a function of their DCS.

In both V2V-only and V2V with R2V configurations, MCORM (A2C) shows superior performance in object perception ratio at short to medium distances, up to approximately 350 m, when compared to eRMLA. This enhanced performance of MCORM (A2C) reflects its effective policies that intelligently filter and disseminate perception data based on potential

occlusions and distance-related perception factors, a capability that eRMLA appears to lack. In contrast, MDRL-CR outperforms MCORM (A2C) across all distances and configurations. Notably, with only V2V communication, MDRL-CR demonstrates a significant improvement in object perception ratio, particularly at critical short distances between 100 m and 200 m, which are essential for the safety of CAVs. This improvement is due to its consideration of objects that, despite being outside the CAVs' FOV, are still relevant and safety-critical in their driving safety coverages, based on their types, classes, and dynamics. When integrating R2V with V2V, both eRMLA and MCORM (A2C) show comparable performance, which tends to decrease in the object perception ratio with increasing distance; however, MCORM (A2C) slightly outperforms at distances larger than 150 m.

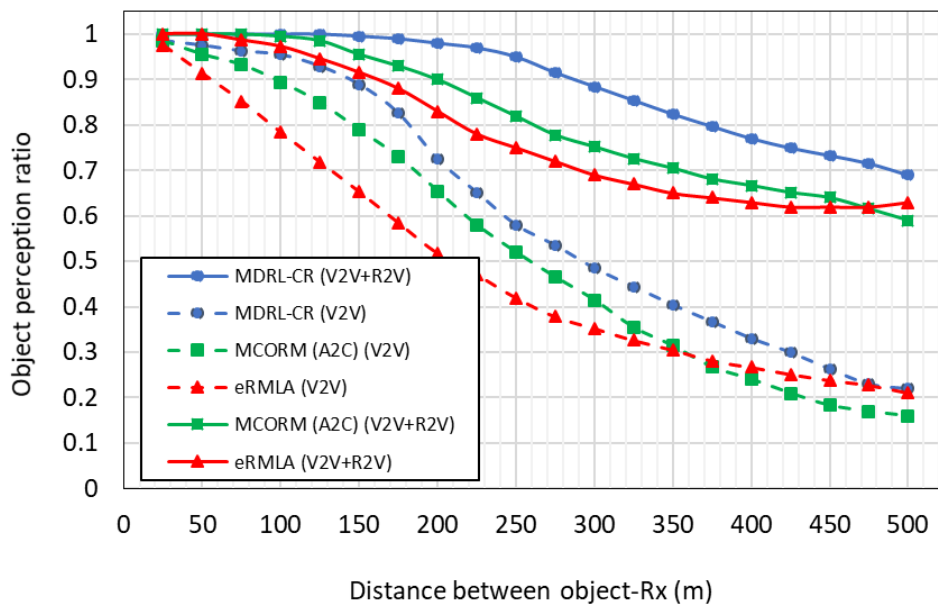


Figure 5.9 Object perception ratio as a function of the distance between the detected object and the CAV receiving it.

The proposed method MDRL-CR consistently maintain a higher object perception ratio, outperforming other methods at distances larger than 150 m, as illustrated in Figure 5.9. This enhanced performance is backed by improvements in network reliability, facilitated by the intelligent use of MCO. These improvements are achieved through the intelligent use of MCO, which allows CAVs to receive optimized CPMs with relevant and safety-critical objects via the CCH. The rest of excluded objects, which have also highlighted their role in augmenting perception awareness, are transmitted through SCH2. This strategy increases network reliability by decreasing the load in CCH, which is typically reserved for critical information exchange for various services, including CAS and CPS.

Figure 5.10 plots the variation of CBR in CCH as a function of 10 CPM generation time intervals for all methods. Operating solely on the CCH, eRMLA and MCORM (A2C) tend to generate a high average load, typically exceeding 60%, with MCORM (A2C) slightly reducing this load by up to 10%. In contrast, MDRL-CR significantly alleviates CBR, showing load

reductions ranging from 20% to 40% over time. This efficiency is achieved by filtering out objects deemed irrelevant or not context-sensitive from the CPMs designated for transmission through the CCH. This is shown in Table 5.2, which depicts the average number of objects included in the CPM at CAVs and RSUs sides. We observe that this number results high at RSUs side due the superior capabilities of their onboard sensors in detecting objects. eRMLA transmits the highest number of objects with limited intelligence, contributing to the increased CBR, as shown in Figure 5.10. In contrast, MCORM (A2C) reduces this number by around 12% and 11% at CAVs and RSUs, respectively. The policies learned by MDRL-CR result in a more substantial reduction by around 41% and 36% at CAVs and RSUs, respectively, in comparison to MCORM (A2C). This suggests that the objects excluded by MDRL-CR are likely non-essential and non-context-sensitive for CAVs.

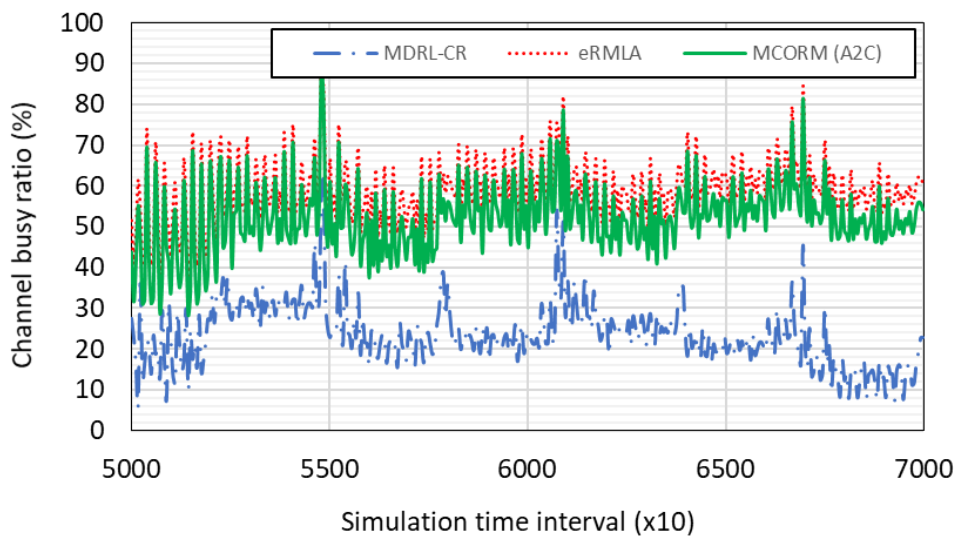


Figure 5.10 The variation of CBR in CCH as a function of simulation time intervals.

Method	Generated by CAVs	Generated by RSUs
eRMLA	30.62	62.68
MCORM (A2C)	26.84	55.98
MDRL-CR	15.78	35.65

Table 5.2 The average number of objects send by CPM for CAVs and RSUs through CCH.

Network reliability and the effectiveness of cooperative perception are also influenced by the latency experienced during the exchange of perception data, particularly those critical for driving safety. We evaluate a latency metric defined as the time difference between when a CPM is generated and when it is received. This metric indicates the freshness of the perception data received via CCH. Table 5.3 presents the average latency times in ms for all methods using

the V2V with R2V configuration. It is observed that eRMLA results in a slightly higher latency, approximately 15% more than MCORM, which can be caused by the greater number of objects transmitted over the CCH. In contrast, MDRL-CR significantly reduces latency by 35% compared to MCORM (A2C). The lower latency communication achieved by MDRL-CR is particularly advantageous in high-density traffic scenario with numerous VRU and Disconnected objects, where CAVs requires timely, effective, and enhanced perception of their environment to fulfill their needs in critical driving situations, thus enabling safer and more efficient operations.

Method	Latency ( <i>ms</i> )
eRMLA	8.75
MCORM (A2C)	7.40
MDRL-CR	4.81

Table 5.3 Latency of CPMs in CCH

## 5.6 Conclusion

This chapter introduced MDRL-CR, an innovative approach designed to enhance the quality of CAP within complex environments with varying conditions. By leveraging the power of advantage actor-critic learning, MDRL-CR enables CAVs and RSUs to learn relevance-aware and context-sensitive policies for the inclusion and transmission of perception data via CPMs. The strength of the proposed approach lies in its comprehensive consideration of various perceptual factors that relate both to the perceived object and the target receiving CAV. The primary aim is to prioritize the transmission of safety-critical objects, considering the current challenges encountered by potential receivers in congested and intricate driving scenarios. The proposed approach is evaluated and compared with recent state-of-the-art methods, based on various key performance indicators. The results demonstrated the proposal's ability to dynamically adjust its policies in real-time, thereby providing CAVs with increased access to safety-critical information in areas of high traffic density and complexity. Furthermore, results underscored the proposal's contribution to network reliability through its intelligent multi-channel operations, which significantly reduced channel load and information latency

While MDRL-CR addresses the challenge of varying conditions among objects and potential receivers during the CPM inclusion process, considering simultaneously the actual congestion level within the network as well as the behavior of DCC is crucial to ensure reliable transmission of perception information from the sender to the receiving CAV. In addition, assuming uniform on-board sensor capabilities for CAVs and RSUs is somewhat unrealistic in real-world scenarios and may negatively impact the effectiveness of their CAP systems.

The next chapter introduces a congestion-aware approach that adapt the inclusion and transmission of objects via CPMs among CAVs and RSUs to the DCC behavior and congestion

level within the network. This could develop more aware and resilient communication strategies for CAVs and RSUs to improve overall quality of their CAP systems.

# Chapter 6

## Asynchronous RL for Congestion-Aware Cooperative Perception in Vehicular Networks Enabled by DCC

### 6.1 Introduction

CAVs and RSUs has significantly enhanced the capability to perceive and interact with the surrounding vehicular environment through the use of onboard sensors. These sensors, however, exhibit variations in their perception capabilities, including difference in detection ranges and FoVs. Contrary to the assumptions of uniform sensor capabilities assumed in the previous chapter, excluding objects that fall outside the sensing coverage of the receiving CAV from transmission could decrease perception awareness and, consequently, reduce the quality and effectiveness of CAP systems. Therefore, the process of including objects in the CPM for transmission must consider for the unique perception capabilities of each receiving CAV.

As vehicular density increases, the complexity of the driving environment correspondingly, necessitate the exchange of a higher number of messages, potentially of increased size. This rise in communication demand can induce congestion within the communication channels, negatively impacting the performance of safety-critical applications and services. The efficacy of CAP systems is fundamentally depending on the reliable and accurate reception of V2Xmessages among CAVs and RSUs. In response to these challenges, the DCC mechanism primarily addresses the challenge of channel congestion across various network layers. At the Access layer, the DCC controls the total amount of messages that a transmitting station can transmit per second [40]. Meanwhile, at the Facilities layer, known as DCC\_FAC, it regulates the generation time of messages by each application or service [42]. This regulation ensures compliance with the DCC Access limit imposed to each station. By leveraging parameters such as the DCC Access limit, message size, and the interval between messages from each application or service, DCC\_FAC estimates the optimal interval and message size for each application or service. This process is guided by the traffic class associated with each service or application as well as the current CBR. The objective is to facilitate efficient and controlled message transmission within the network, thus addressing the dual challenges of maintaining high-quality perception awareness while mitigating the adverse effects of channel congestion.



## 6.2 Impact of DCC on CAP Quality

DCC algorithms can modify the rate or power of information transmission or even drop packets associated to CPMs, changing the transmission strategies of V2X messages [128]. This regulation directly influences the effectiveness of CP [129] and [130], as safety-critical applications in autonomous driving impose stringent constraints on delays. Specifically, the latency between the generation and transmission of CPMs cannot exceed 100 ms [127]. The work in [131] studied the impact of the ETSI ITS G5 network on cooperative perception, examining both the CPM generation rules and the effect of transmission intervals on communication channels. They evaluated the impact of DCC on cooperative perception by defining a constrained environment for hundreds of vehicles. Their analysis underscored that the volume of perception data generated for transmission could easily overload the communication channels, leading to increased latency and, consequently, diminished accuracy [132]. Further investigations [133] into the effects of DCC on CAP systems in dense traffic scenarios revealed that high channel loads increase the risk of data packet collisions. This results in reduced transmission efficiency, increased communication delays, and compromised accuracy of objects in CPMs.

To address the challenges, the object inclusion and transmission process of CPMs must adapt to the DCC restriction to main the quality of CAP systems. This chapter introduces an evolved optimization problem and learning paradigm for CAVs and RSU, specifically tailored to align with DCC operational mode, particularly at the Facilities layer. while the original MDRL-CR approach proposed in the previous chapter considers various factors, including perception capabilities and the driving context of the receiving CAVs, to determine the inclusion of objects in CPMs, its performance may shift with the introduction of DCC.

The newly refined MDRL-CR-DCC approach of this chapter adapts the inclusion criteria and sizes of CPMs to align with the specific limitations set by DCC, which are based on the current availability of channel resources. This adaptation leverages the A3C algorithm [124] and [134], ensuring seamless integration with DCC's operational dynamics. The A3C architecture empowers CAVs and RSUs to independently and asynchronously train on local models, while simultaneously contributing to the refinement of a shared global model. This process involves each of them interacting with the environment, computing gradients, and transmitting updates to a central network. The global network, in turn, assimilates these insights to update the parameters of it model and subsequently redistributes them across the network to enhance the collective decision-making process. By adopting this strategy, the MDRL-CR-DCC framework not only improves the learning process but also significantly enhances the capability of its model to generate efficient and effective outcomes under a variety of demanding scenarios simultaneously. This ensures that the system remains Resilient and adaptive, capable of optimizing CPM selection and transmission strategies in the face of fluctuating environment and network conditions under DCC restrictions.

## 6.3 Congestion-Aware Problem Formulation and Modeling

This section formulates and model the underlying optimization problem of MDRL-CR-DCC.

### 6.3.1 Problem Formulation

Leveraging the network architecture, components, and communication scenarios presented in Chapter 6, this section introduces a refined approach to problem formulation for the MDRL-CR approach. The essence of this enhanced formulation lies not only in adhering to the objectives of developing relevance-based and context-aware object selection strategies for inclusion in the CPM, tailored to the specific needs of each receiving CAV and RSU in their respective contexts as outlined by Equation (5.5), but also in addressing a critical oversight. The original MDRL-CR framework did not adequately account for the constraints imposed by the DCC mechanism, particularly regarding message size limitations. Such constraints significantly impact the quality and effectiveness of CAP systems by potentially hampering the transmitting station's ability to disseminate its CPM due to the increased size of the message and current congestion level in the network.

The DCC mechanism operates across various layers to mitigate these challenges. Specifically, at the Facilities layer, the DCC\_FAC component plays a pivotal role by gauging the available channel resources, as determined by the DCC\_ACC, across each radio channel. This assessment is facilitated by the DCC\_CROSS at the Management layer, which ensures seamless interactions with DCC elements from other layers. The operational protocol at this stage, which is applied to each CPM generation time, analyses the message length  $L_{i,j}$  of the recently generated message and estimates the average message length  $\overline{L}_{i,j}$  for application/service  $j$  and traffic class  $i$ .

$$\overline{L}_{i,j} = w * L_{i,j} + (1 - w) * \overline{L}'_{i,j}, \quad (6.1)$$

where  $\overline{L}'_{i,j}$  represents the previously estimated average message for application/service  $j$  and traffic class  $i$ . In the absence of prior messages,  $\overline{L}'_{i,j}$  defaults to  $L_{i,j}$ . The coefficient  $w$  is set to 0.1, as suggested by [42].

To overcome the limitations introduced by the DCC due to current congestion levels, the MDRL-CR-DCC framework could consider the channel resources reserved for transmission during the CPM preparation and dissemination process. This adjustment could be seamlessly integrated by exploiting the estimated message size  $\overline{L}_{i,j}$  determined by the DCC at Facilities layer, as follows:

$$\text{maximize} \quad \sum_{r \in C_i(t)} DCS_r(t) \sum_{o \in M_i(t)} (1 - F_{o,r}(t) * b_{o,r})^{1-P(o)}$$

Subject to: (6.2)

$$C1: \quad M_i(t) \subseteq T_i(t)$$

$$C2: \quad \sum_{o \in M_i(t)} \text{size}(o) \leq \overline{L_{v,j}}$$

$$C3: \quad b_{o,r} \in \{0,1\}$$

$$C4: \quad t_{av}, t_{ar} < t < 2 * (t_{av}, t_{ar})$$

where  $\text{size}(o)$  denotes the size of object  $o$ , and  $b_{o,r}$  is binary variable that indicates whether the object is within the sensing and of FoV receiving CAV. Constraint C2 ensures that the size of the CPM does not exceed the maximum size limit  $\overline{L_{v,j}}$  imposed by the DCC. The idea behind Equation 6.2 lies in giving more attention to objects that are outside the sensing range and FoV of the receiving CAV, aiming to enhance situational awareness by incorporating critical objects into the CPM without surpassing the maximum allowable size stipulated by the DCC.

## 6.2.1 System Model

The environment state of a given CAV or RSU of index  $i$  denoted as  $s_i(t)$ , encompasses the set of receiving CAVs  $E_i(t)$  in its communication coverage at time  $t$ . This set is basically defined in Equation (5.7), with each CAV also includes a FoV matrix  $FoV$ , mapping out the area it covers using its onboard sensors. Figure 6.1 depicts a possible FoV of a receiving CAV. Considering a maximum sensing range  $m$  and a FoV of  $360^\circ$ , the cells within the sensor coverage are denoted by a value of 1, whereas cells falling outside are denoted by 0.

Moreover, the environment state also integrates additional crucial parameters such as the list of perceived objects  $T_i(t)$ , the current CBR  $CBR_i$ , and the CPM generation frequency  $freq_i$ . These elements aid in the estimation of the CPM size, as specified by [42]. The environment state can be denoted by:

$$s_i(t) = \{E_i(t), T_i(t), CBR_i, freq_i\} \quad (6.3)$$

The action  $a_i(t)$  of each CAV and RSU of index  $i$  is the same as defined for MDRL-CR. However, the reward should be meticulously designed to account for the implications of exceeding the DCC-imposed message size limits and to prioritize information from regions not covered by onboard sensors of the receiving CAVs. The reward  $r_i(t)$  is given as follows:

$$r_i(t) = \sum_{r \in C_i(t)} DCS_r(t) * \sum_{o \in M_i(t)} (1 - F_{o,r}(t) * b_{o,r})^{1-P(o)} * N_i(t) \quad (6.4)$$

where  $b_{o,r}$  in this case corresponds to the binary value of the cell region of the receiving CAV's FoV matrix that encompasses the object  $o$ .  $N_i(t)$  quantifies the impact of the CPM size regarding DCC restrictions, defined by:

$$N_i(t) = \begin{cases} \frac{1}{\sqrt{L - \text{size}(M_i(t)) + 1}}, & \text{size}(M_i(t)) \leq L \\ \frac{1}{\text{size}(M_i(t)) - L + 1}, & \text{otherwise} \end{cases} \quad (6.5)$$

The refined reward defined for MDRL-CR-DCC plays a crucial role in guiding CAVs and RSUs toward making optimal decisions, particularly regarding the selection of relevant objects and the sizing of CPMs. Its core is strategically designed to penalize the generation of CPMs that surpass the size limit set by the DCC, aiming to deprioritize CPMs that are larger than the DCC's size limit which are likely to be blocked from transmission within the network. Such limitations not only impede the transmission of vital information among CAVs and RSUs but also compromise the quality CAP systems, increasing latency and decreasing perception awareness.

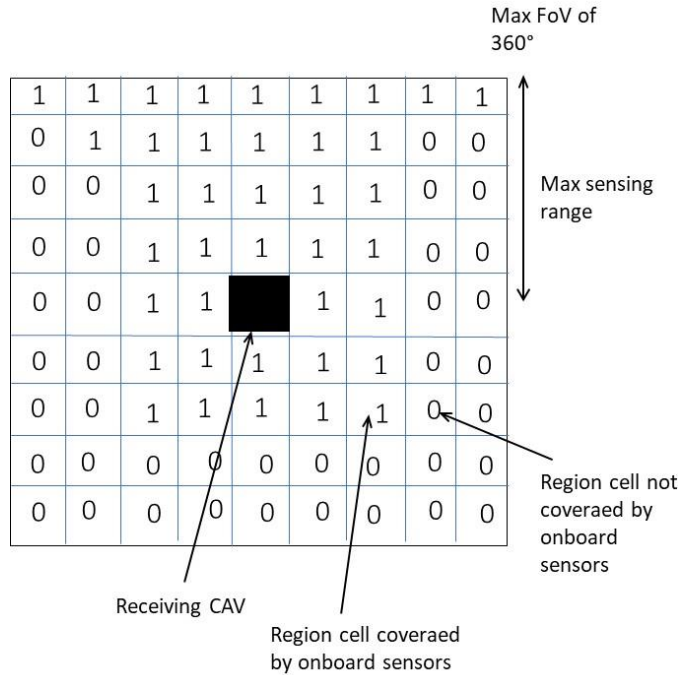


Figure 6.1 Possible FoV of a CAV

To overcome these issues, the reward is designed to give more attention to actions that result in the generation of optimal CPMs within the acceptable size range which are likely to be transmitted successfully. Through this design MDRL-CR-DCC aims to enhance the efficiency and efficacy of communication within vehicular networks enabled by DCC. By optimizing the selection and sizing of CPMs, it ensures that critical information is reliably shared among CAVs

and RSUs, thereby improving cooperative perception accuracy and, ultimately, contributing to safer and more efficient driving.

## **6.3 Asynchronous RL for Congestion-aware cooperative perception**

This section presents asynchronous RL as an efficient method, allowing MDRL-CR-DCC to train efficiently while aligning with the DCC operational mode.

### **6.3.1 Asynchronous RL in DCC-Enabled Vehicular Networks**

#### **6.3.1.1 Motivation and Challenges**

The fundamental principle behind the DCC mechanism empowers each transmitting station within a vehicular network to independently adjust its transmission strategy in response to real-time network conditions. The operation of DCC does not require any centralized controller or coordination among transmitting stations. Instead, each station employs predefined algorithms that determine how to adjust transmission parameters based on local network conditions. These algorithms are designed to ensure that all stations collectively contribute to maintaining optimal network performance without direct interaction or coordination.

The original MDRL-CR employs a distributed architecture of the A2C algorithm in a MARL environment. This architecture enables CAVs and RSUs to improve the quality and effectiveness of their CAP systems by selecting and transmitting relevant and safety-critical objects through optimized CPMs in a synchronized manner [135]. This means that all CAVs and RSUs should select objects and transmit CPMs at every corresponding message generation time. However, with DCC, a group of CAVs or RSUs may not be able to exchange messages at the current message generation time due to elevated congestion levels in the network. This appears to contradict the principle of the original MDRL-CR when considering DCC.

#### **6.3.1.2 Toward Asynchronous RL for DCC-Enabled Networks**

Synchronous and asynchronous RL are two approaches used to update the parameters of a learning model. In synchronous reinforcement learning, updates to the model's parameters occur in a coordinated manner, typically after all agents in a distributed system have completed a round of interactions with the environment. This means that all agents wait for each other before updating their parameters, ensuring consistency across the learning process. On the other hand, asynchronous reinforcement learning allows agents to update their parameters independently and asynchronously, without waiting for other agents to finish their interactions. This approach often leads to faster learning as agents can make use of more recent experiences to update their models. The choice between synchronous and asynchronous RL depends on the specific requirements of the problem at hand, including considerations such as computational

resources, communication overhead, and the desired trade-off between speed and stability. In the context of our problem, this means each CAV and RSU independently trains and updates a global model or policy, guided by individual observations, actions, and obtained rewards. This mode of operation diverges significantly from the synchronous learning approach utilized in the original MDRL-CR, aligning more closely with the dynamic and complex nature of vehicular communication under congestion scenarios.

Implementing asynchronous RL for CPM selection in DCC-enabled networks aligns with the requirements for a scalable, efficient, and adaptive communication system. It addresses the inherent complexities and dynamic nature of vehicular communications under congestion. Through independent learning experiences, CAVs and RSUs can uncover and contribute to a broader collective intelligence, leading to the discovery of augmented capabilities for intricate scenarios. This process not only enhances the development of more effective communication strategies but also bolsters the network's ability to navigate novel or unforeseen challenges.

Techniques such as the Asynchronous Advantage Actor-Critic (A3C) have demonstrated considerable success in complex environments by leveraging the strengths of parallel processing and asynchronous updates [124].

## 6.3.2 MDRL-CR-DCC Learning Algorithms

This section details the learning algorithm of MDRL-CR-DCC, employing A3C as a core learning principle.

### 6.3.2.1 Introduction and Architecture of A3C

The A3C learning algorithm [124] is an improvement of the A2C algorithm [135], allowing multiple worker threads to interact with the environment and asynchronously train the worker threads, as defined in Figure 6.2.

When the maximum number of training iterations is reached, each worker thread computes locally its gradient and sends it to the global network. To ensure that each worker thread can share the same policy, the global network updates the global parameters and distributes them to each worker thread. This strategy improves the learning process and ensures efficient outputs in various challenging situations.

### 6.3.2.2 Architecture and Learning Algorithm for Congestion-Aware

The architecture of MDRL-CR-DCC is depicted in Figure 6.3. Each CAV and RSU employs a pair of DNNs:  $\pi^v$  and  $Q^v$  for CAVs, and  $\pi^{rsu}$  and  $Q^{rsu}$  for RSUs. These networks serve to represent the policy network and value network of each entity.

The learning algorithm can be divided into several key stages is illustrated in Algorithm 6.1:

- Initialization Stage: Global actor and critic networks each with associated learning episodes and steps per each episode are established for both CAVs and RSUs on an edge

computing device, laying the foundation for collective learning and strategy optimization across the network.

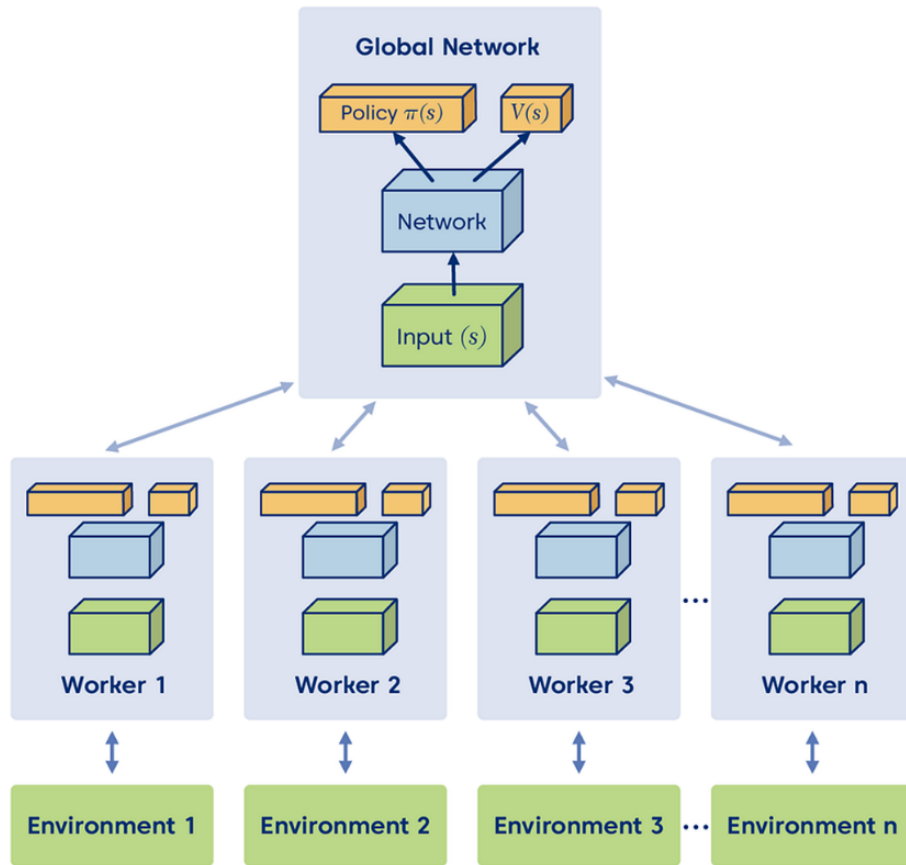


Figure 6.2 Illustration of the A3C principle [124]

- **Parallel and Asynchronous Learning Stage:** Each CAV and RSU engages in individual learning sessions as depicted by Algorithm 5, leveraging their local DNNs to make decisions and learn from outcomes in real-time. These sessions are conducted in parallel and asynchronously, ensuring each CAVs and RSUs act according to the congestion level within its network based on the DCC.
- **Experience Collection and Storage Stage:** As CAVs and RSUs interact with the environment, they collect data on their experiences, including the state of the network, actions taken, and resulting outcomes. This data is stored in an experience buffer, forming a dataset from which the DNNs can learn and improve its outcomes.
- **Local training and Updates Stage:** Using the stored experiences, each CAV and RSU periodically updates its DNNs. This involves computing gradients, adjusting weights, and improving the policy and value estimation functions based on observed rewards and outcomes, as illustrated by Algorithm 6.

- Global Model Updates Stage: The local updates from all CAVs and RSUs are combined to update their associated global actor and critic networks on the edge computing device. This means that critical insights and improvements from individual local learnings are periodically integrated into the global models and then shared back to CAVs and RSUs to update their local models, reflecting the new global knowledge.

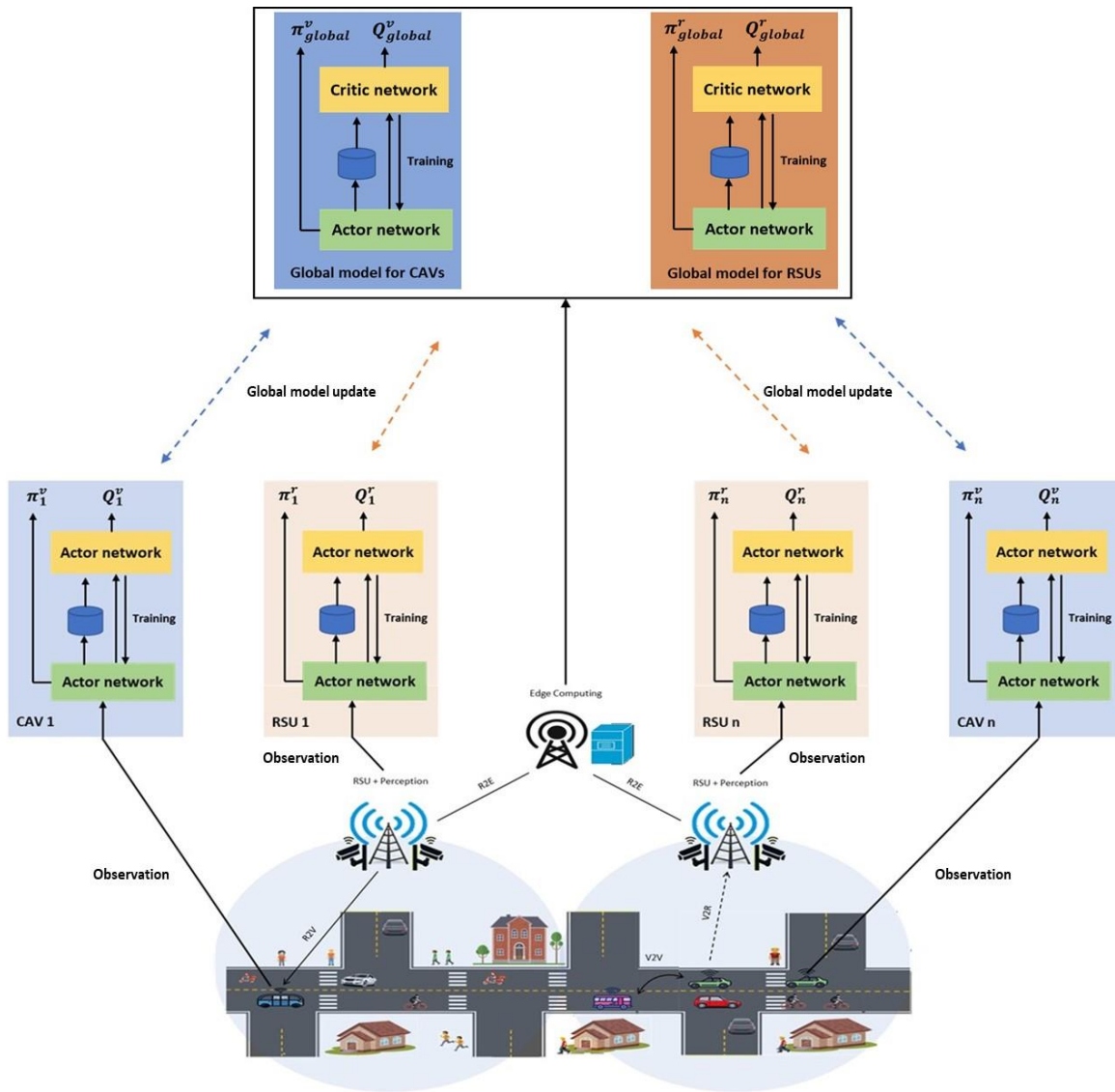


Figure 6.3 Architecture of MDRL-CR-DCC based on A3C



---

**Algorithm 5:** The training of MDRL-CR-DCC based on A3C algorithm

---

- 1: **Initialization:**
  - 2: | A global actor network  $\pi_{global}^v$  with parameters  $\theta_{global}^v$  and critic network  $Q_{global}^v$  with parameters  $\theta_{global}^v$  for CAVs on the edge computing device.
  - 3: | A global actor network  $\pi_{global}^{rsu}$  with parameters  $\theta_{global}^{rsu}$  and critic network  $Q_{global}^{rsu}$  with parameters  $\theta_{global}^{rsu}$  for RSUs on the edge computing device.
  - 3: |  $N_{ep}V, N_{ep}R$ : number of episodes of the global CAV and RSU models on the edge.
  - 4: |  $N_{steps}V, N_{steps}R$ : number of steps by episode of the global CAV and RSU models on the edge
  - 5: **Output:**
  - 6: | Trained model for each CAV and RSU
  - 7: **Begin:**
  - 8: | **Do in parallel and asynchronously:**
  - 9: | | Train (MARL-CAV-DCC,  $N_{ep}V, N_{steps}V, \pi_{global}^v, \theta_{global}^v, Q_{global}^v, \theta_{global}^v$  )
  - 10: | | Train (MARL-RSU-DCC,  $N_{ep}R, N_{steps}R, \pi_{global}^{rsu}, \theta_{global}^{rsu}, Q_{global}^{rsu}, \theta_{global}^{rsu}$  )
  - 11: | **end do**
  - 12: **end**
-

---

**Algorithm 6:** Train (sub\_system\_X,  $N_{ep}^X$ ,  $N_{steps}^X$ ,  $\pi_{global}^x$ ,  $\theta_{global}^x$ ,  $Q_{global}^x$ ,  $\theta_{global}^{x-}$ )

---

```

1:  Initialization:
2:  foreach station of type X (CAV or RSU)
3:      An actor network  $\pi^x$  with parameters  $\theta^x \leftarrow \theta_{global}^x$ ,
4:      A critic network  $Q^x$  with parameters  $\theta^{x-} \leftarrow \theta_{global}^{x-}$ 
5:      An experience buffer  $\mathcal{D}^x$ 
6:       $N_{epLocal}^X$ : number of local training episodes
7:       $N_{stepsLocal}^X$ : number of steps per local learning episode
8:      Discount factor  $\gamma$  and learning rate  $\alpha$  in  $[0,1]$ 
9:  End foreach
10: Begin:
11:    $ep_{local} \leftarrow 0$ ,  $ep_{global} \leftarrow 0$ ,  $t_{local} \leftarrow 0$ ,  $t_{global} \leftarrow 0$ 
12:   while  $ep_{local} < N_{epLocal}^X$  and  $ep_{global} < N_{ep}^X$  do
13:       while  $t_{local} < N_{stepsLocal}^X$  and  $t_{global} < N_{steps}^X$  for each agent of type X do
14:           Build  $s^x(t)$ , select  $a^x(t)$ , and get  $r^x(t)$ 
15:           Store  $e^x(t) = ((s^x(t), a^x(t), r^x(t), s^x(t+1)))$  into  $\mathcal{D}^x$ 
16:            $t_{local} \leftarrow t_{local} + 1$ ,  $t_{global} \leftarrow t_{global} + 1$ 
17:       End while
18:       If  $t_{local} == N_{stepLocal}$  do
19:            $\theta^{x-}, \theta^x \leftarrow \mathbf{update\_local\_model}(\mathcal{D}^x, \theta^{x-}, \theta^x)$ 
20:            $t_{local} \leftarrow 0$ 
21:       else
22:            $t_{local} \leftarrow t_{local} + 1$ 
23:       endif
24:       If  $t_{global} == N_{stepGlobal}$  do
25:            $\theta^{x-}, \theta^x \leftarrow \mathbf{udapte\_local\_model}(\text{sub\_system}_x, \theta_{global}^x, \theta_{global}^{x-})$ 
26:            $t_{global} \leftarrow 0$ 
27:       else
28:            $t_{global} \leftarrow t_{global} + 1$ 
29:       endif
30:        $ep_{local} \leftarrow ep_{local} + 1$ 
31:        $ep_{global} \leftarrow ep_{global} + 1$ 
32:   end while
33: end

```

---

---

**Algorithm 7:** update\_local\_model( $\mathcal{D}^x, \theta^x, \theta^{x^-}$ )

---

1: **Begin**

2: Sample  $(s^x, a^x, r^x, s^x + 1) \sim U(\mathcal{D}^x)$  of size  $n_e$

3: Compute the critic loss:

$$Loss(\theta^{x^-}) = \left(\frac{1}{n_e}\right) \sum_{t=0}^{n_e-1} \left(r^x(t) + \gamma Q_{\theta^{x^-}}^x(s^x(t+1)) - Q_{\theta^{x^-}}^x(s^x(t), a^x(t))\right)^2$$

4: Update the weights of the critic model:

$$\theta^{x^-} = \theta^{x^-} + \alpha * Loss(\theta^{x^-})$$

5: Compute the gradient of the policy model:

$$\nabla_{\theta^x} J(\theta^x) = \sum_{t=0}^{|\tau_i|-1} \nabla_{\theta^x} \log \pi_{\theta^x}^x(a^x(t)|s^x(t); \theta^x) A(s^x(t), a^x(t))$$

6: Update the weights of the actor model:

$$\theta^x \leftarrow \theta^x + \beta * \nabla_{\theta^x} J(\theta^x)$$

7: **Return** ( $\theta^x, \theta^{x^-}$ )

8: **End.**

---

---

**Algorithm 8:** udapte\_global\_model(sub\_system\_X,  $\theta_{global}^x, \theta_{global}^{x^-}$ )

---

1: **Begin**

2: Receive the list of gradients  $\{\nabla_{\theta^{x,1}} J(\theta^{x,1}), \nabla_{\theta^{x,2}} J(\theta^{x,2}), \dots, \nabla_{\theta^{x,n_a}} J(\theta^{x,n_a})\}$  from  $n_a$  actor models of type X

3: Receive the list of gradients  $\{Loss(\theta^{x^-,1}), Loss(\theta^{x^-,2}), \dots, Loss(\theta^{x^-,n_c})\}$  from  $n_c$  critic models of type X

4:

$$\theta_{global}^x = \theta_{global}^x + \left(\frac{1}{n_a}\right) \sum_{i=1}^{n_a} \left(\nabla_{\theta^{x,i}} J(\theta^{x,i})\right)$$
$$\theta_{global}^{x^-} = \theta_{global}^{x^-} + \left(\frac{1}{n_c}\right) \sum_{i=1}^{n_c} \left(Loss(\theta^{x^-,i})\right)$$

5: **Return** ( $\theta_{global}^x, \theta_{global}^{x^-}$ )

6: **End.**

---

## 6.4 Evaluation Results

To evaluate MDRL-CR-DCC, we conduct simulations using Artery and SUMO. This simulation spans 70000 time interval, including 5000 training episodes for the global model on the Edge computing device which each episode comprises 10 steps.

The evaluation scenario features multiple RSUs and a large number of vehicles with diverse dimensions, types (including both connected and disconnected), and classes, as well as road

obstacles on a selected map from OpenStreetMap. Each CAV and RSU has its own number of episodes and steps by episode. CAVs are equipped with radar, lidar, and camera sensors, each offering different FoV and sensing ranges. In contrast, the RSUs are designed with a uniform sensing horizon of up to 300 meters. Table 6.1 summarizes the configurations used for this evaluation.

Parameter	Value
Communication Profile	ITS-G5
CAM Generation Time (default)	$t_{av} = t_{ar} = 100ms$
CPM Generation Time (default)	$t_{pv} = t_{pr} = 150ms$
DCC	Reactive
Data Profile (CAM, CPM)	(1,2)
Primary Channel	CCH
Secondary Channel	SCH2
Tertiary Channel: Used for V2R, V2E, and R2E exchanges	SCH3
Transmission power (default)	23 dBm
Data rate (default)	6 Mbps
CAV Density	250 CAV/lane/km
Sensors	Radar,Lidar, Camera
FoV and sensing range of CAVs	Varied (30 to 100 m; 40 to 360°)
FoV and sensing range of RSU	360°, 300 m
Average vehicle size (length, width)	(4m, 2m)
Object priorities ( $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ )	(1, 0.75, 0.5, 0.25)
Safety coverage range	100 m
Simulation time	70000 time slots
Number of episodes of the global model	5000
Number of steps by a single episode of the global model	10

Table 6.1: Parameters used for simulation of MDRL-CR-DCC

CAVs and RSUs are configured to exchange CAMs and CPMs at default intervals of 100 ms and 150 ms, respectively, based on the ETSI ITS-G5 standard. Similarly, CAMs and CPMs optimized and generated by MDRL-CR-DCC are exchanged through the CCH between CAVs and RSUs via V2V and R2V communication scenarios. The DCC is applied only on the CCH. To ensure that CAMs are sent before CPMs, thus satisfying the constraint of the optimization problem that MDRL-CR-DCC aims to solve, we assign them a lower DCC profile than CPMs. Additional interactions occur through SCH2 and SCH3, as previously described in Chapter 5 for MDRL-CR without congestion control. Each generated CPM includes all essential headers and containers, specifically one ITS PDU header, one MC, and one OVC, resulting in 121

Bytes. Moreover, for every detected object, one POC (35 Bytes) is included in the CPM. Additionally, a SIC (35 Bytes) is added to the CPM every second. The size of the CPM is dynamically calculated by the transmitting CAV or RSU, based on the cumulative sizes of the included containers.

### 6.4.1 Reward and Loss Variation

Figure 6.4 and Figure 6.5 depict the average reward and loss variation of CAVs and RSUs using MDRL-CR-DCC, based on both the A2C and A3C algorithms, as a function of the training episodes. In the case of A2C, we assume that all vehicles have the same number of episodes as the global model of A3C.

The purpose of this comparison is to illustrate that synchronous learning using A2C under DCC restrictions may not perform well and might not achieve a larger number of training episodes, as indicated by the green curve. This reflects the congestion level in the network, where a group of CAVs and RSUs cannot complete a set training episode due to the reduced message generation frequency imposed by the DCC, resulting in a decrease in the number of episodes completed by the agent.

However, with asynchronous learning based on the A3C algorithm, even CAVs and RSUs that cannot execute training steps still receive an updated copy of the global model after each episode, enabling them to generate and exchange optimized CPMs.

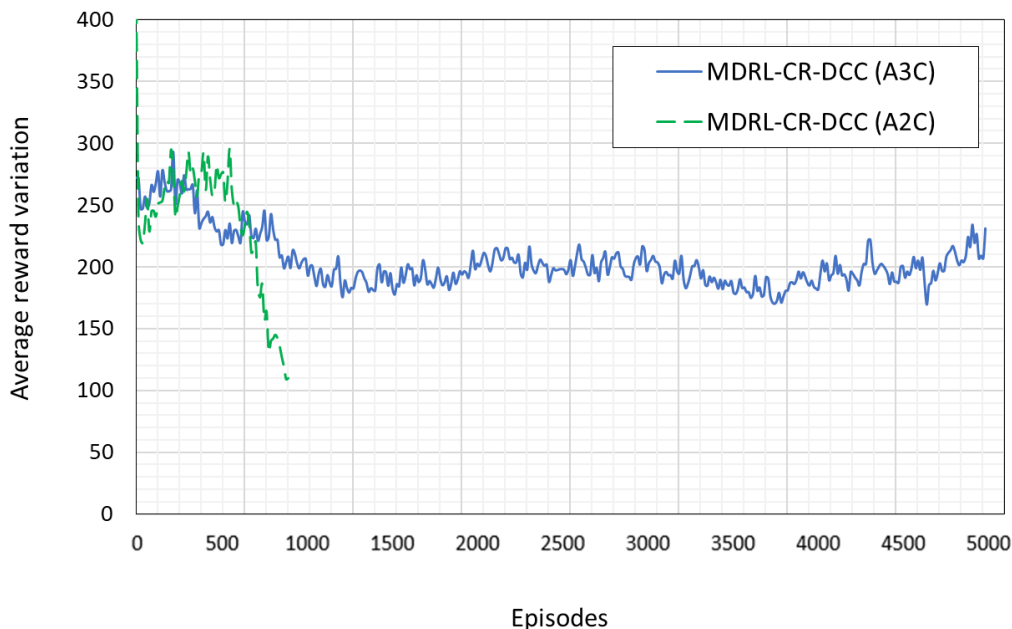


Figure 6.4 Average reward variation as a function of training episodes of the global model

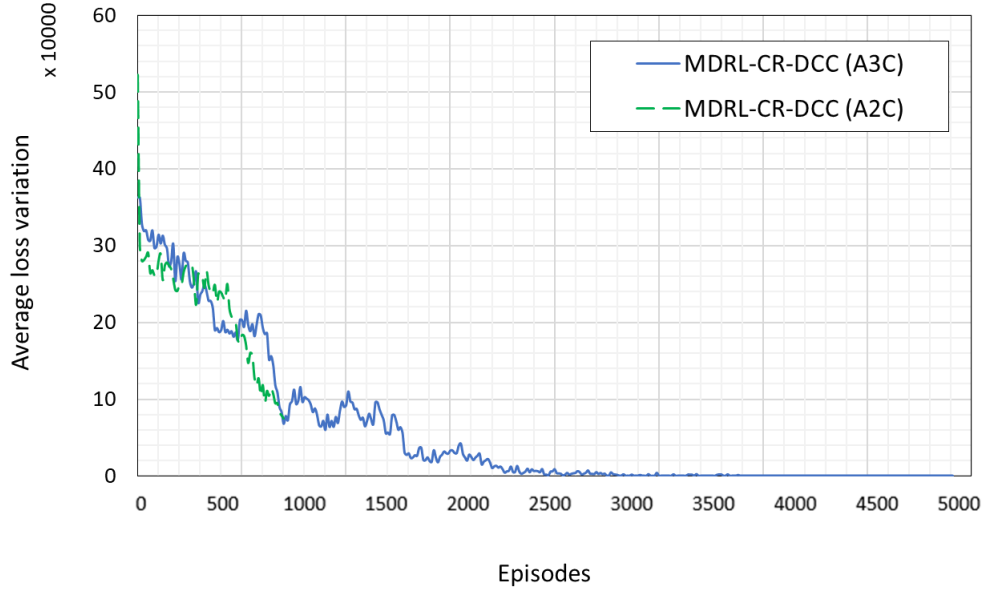


Figure 6.5 Average loss variation as a function of training episodes of the global model of A3C

## 6.4.2 Evaluation of CAP quality

We compare the performance of MDRL-CR-DCC based on the A3C algorithm with the eRMLA method [90], after the last global model has reached the final update episode (5000 episode). Figure 6.6 depicts the variation in the expected size imposed by the DCC, the size of the CPM detected objects, and the size of the CPMs generated by MDRL-CR-DCC and eRMLA. The size of all detected objects results in higher values, up to 1400 bytes, due to the high vehicle density in the considered scenario. On the other hand, the expected value imposed by the DCC is approximately 40% lower than this, highlighting the higher congestion level in the network. The size of the CPM generated by eRMLA exceeds the expected size by values varying between 200 to 600 bytes. However, MDRL-CR-DCC keeps the CPM size close to the expected size thanks to the modeling of the reward functions, which penalize CPMs with sizes that exceed the expected size more than those with sizes that are less than expected. This allows to remain as close as possible to the expected size, ensuring that the CPMs are generated and transmitted without blocking caused by the increased size and DCC restrictions.

The variation in CPM size by each model has its impact on CBR, which directly reflects message rate and time between two transmissions, according to the DCC state machine of the DCC reactive approach. This is depicted in Figure 6.7, which plots the CBR variation of MDRL-CR-DCC and eRMLA using V2V and R2V exchanges in CCH as a function of simulation time steps after the training phase. Figure 6.7 shows that the excess in CPM size achieved by eRMLA results in a higher CBR value compared to MDRL-CR-DCC. The CBR value achieved by eRMLA is on average 43%, falling into state Active 4 of the DCC reactive state machine, allowing a message rate of 2.5 Hz and a waiting time for the next generation message of 200 ms. On the other hand, MDRL-CR-DCC decreases the CBR value by

approximately 20% on average, increasing the generation frequency of CPMs to 10 Hz and reducing the waiting time to 60 ms.

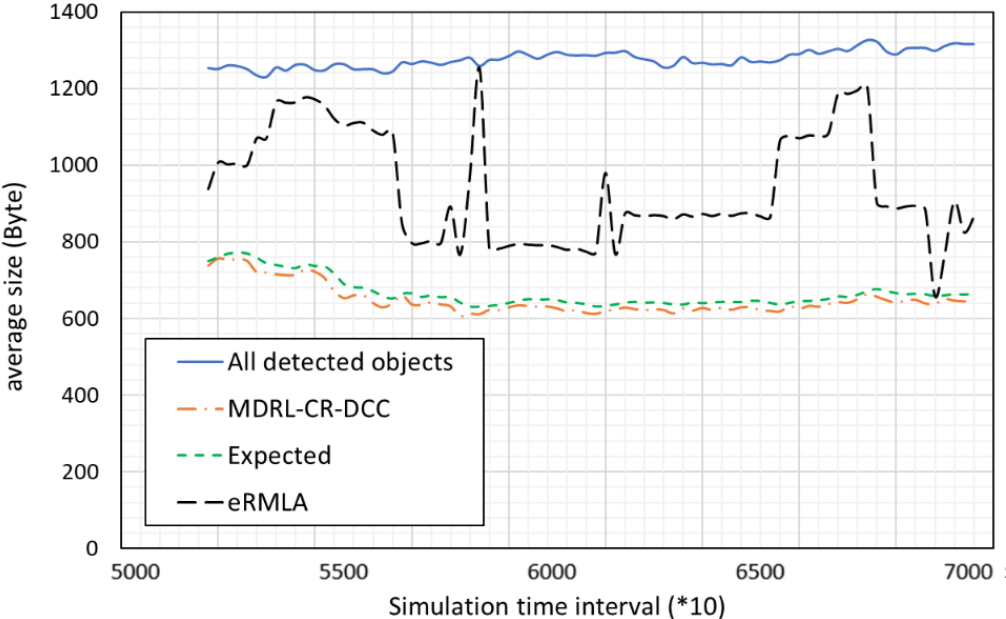


Figure 6.6 CPM size

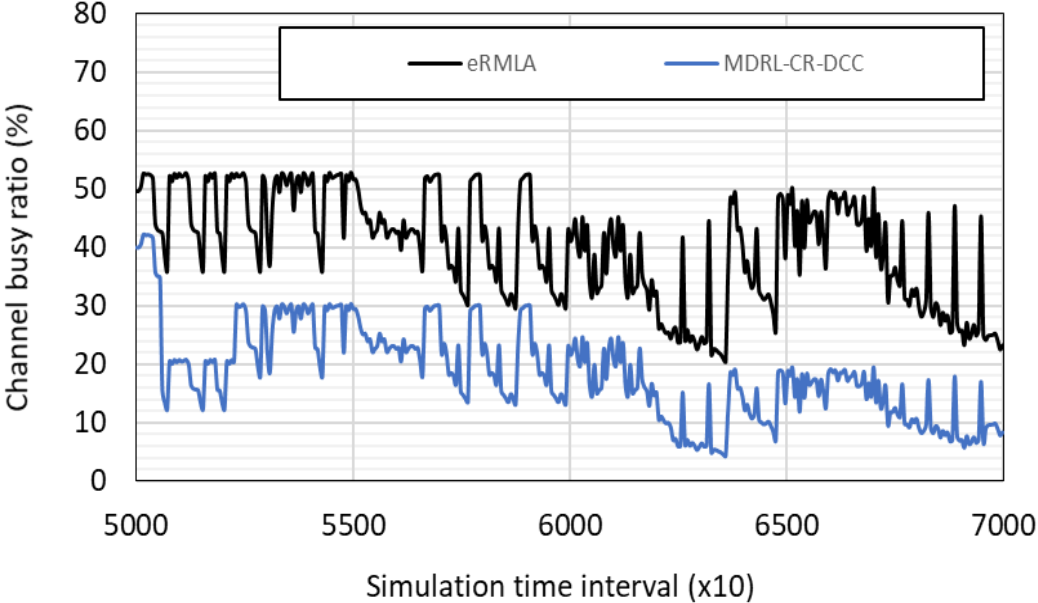


Figure 6.7 The CBR as a function of simulation time interval

The increased number of generated CPMs has enhanced the levels of CP awareness for CAVs based on communications through V2V and R2V performed on CCH, as depicted in Figure 6.8. CP awareness is plotted as a function of the distance between the detected object and the CAV receiving it. This metric decreases significantly with distance increase. This is because the objects are being distant from the receiving CAV, falling out of their varied sensing coverages and FoVs, as well as due to the impact of decreased transmission power imposed by the DCC TPC algorithm. MDRL-CR slightly outperforms eRMLA at short distances up to 100 m. Furthermore, it significantly increases CP awareness compared at medium and large distances larger than 100 m.

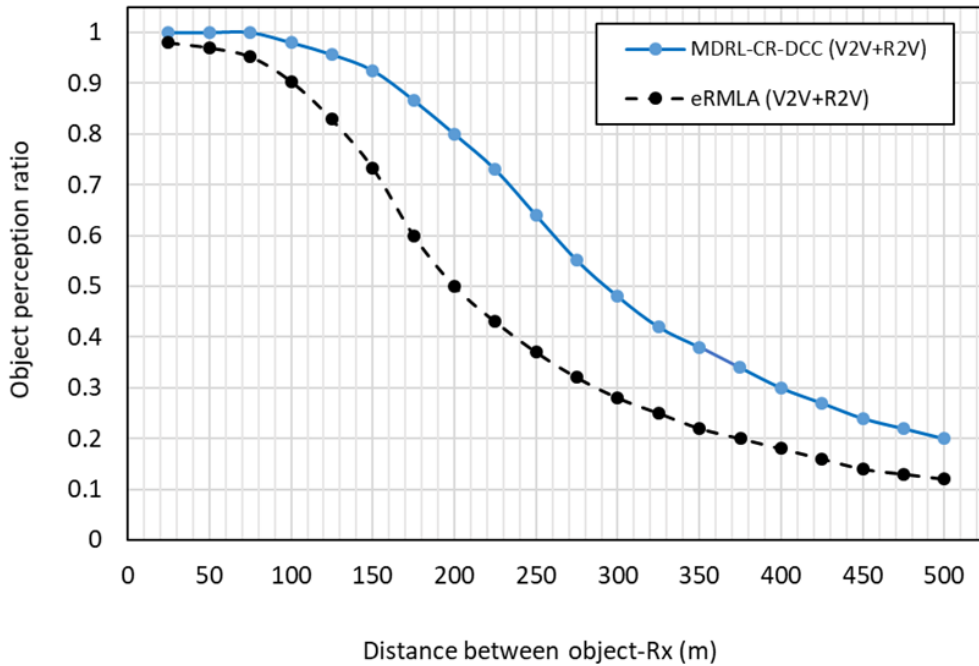


Figure 6.8 The CP awareness as a function of the distance between the detected object and the receiving CAV

## 6.5 Conclusion

This chapter introduced MDRL-CR-DCC, a refined version the MDRL-CR approach proposed in Chapter 5. MDRL-CR-DCC enhances the original approach by incorporating a dynamic adaptation mechanism to the fluctuating congestion levels within vehicular networks during the selection and transmission of CPMs by CAVs and RSUs. This advancement is achieved through a sophisticated modeling that simultaneously considers the relevance of objects, the evolving context and challenges faced by potential receiving CAVs, and the prevailing congestion conditions within the network. The power of MDRL-CR-DCC lies in its adoption of asynchronous learning principles, meticulously designed to align with the DCC operation mode. This allows for each CAV and RSU to asynchronously and concurrently learn



from their environment, facilitating a more robust and decentralized decision-making process that is responsive to real-time network conditions and requirements.

# Chapter 7

## Conclusion and Future Research

### 7.1 Conclusion

Throughout this thesis, we have dedicated our efforts to developing mechanisms to improve the resilience of CAP systems for CAVs and RSUs. Our thesis started with an introduction to the fundamental concepts of C-ITS, where we discussed its key components, reference architectures, and focused on areas such as machine learning, communication technologies, edge computing and services.

In Chapter 3, we delved into the dynamic and evolving realm of CAP in C-ITS, highlighting its potential to enhance road safety and efficiency. The chapter emphasized the necessity for resilience in these systems to effectively navigate a variety of adverse conditions and challenges, particularly focusing on the reliability of information and communication technologies. This discussion highlighted the critical need for consistent and reliable sharing of perception information under challenging conditions, demonstrating how this could enhance the capabilities of onboard sensors and overall environmental awareness.

Chapter 4 introduced V2V-RMM, a novel approach utilizing distributed multi-agent DRL to minimize the transmission of redundant perception information among CAVs. This approach focuses on optimizing the selection of objects for inclusion in CPMs and has been shown to significantly improve network reliability by mitigating redundant information. This improvement allowed CAVs to receive further relevant information that appeared to be lost or not considered for transmission, leading to an increased level of environmental awareness at safety-critical distances of less than 200 meters, compared to existing approaches. Although V2V-RMM is effective in improving CAP systems by increasing environmental awareness among CAVs, it revealed areas for improvement, particularly in object prioritization and the adaptation of strategies to diverse receiver contexts.

Chapter 5 introduced the second contribution MDRL-CR, an enhanced approach designed to improve the quality of CAP within complex environments with varying conditions. By leveraging the power of advantage actor-critic learning, MDRL-CR enabled CAVs and RSUs to learn relevance-aware and context-sensitive policies for the inclusion and transmission objects via CPMs, tailored to each receiving CAV's specific environmental context. The obtained results showcased the proposal's ability to provide CAVs with increased safety-critical information in heightened traffic density and complexity. Furthermore, MDRL-CR enhances

CAP quality, notably reducing CBR by 20% to 40%, and decreasing the communication latency by approximately 35% compared to the existing works.

Chapter 6 addressed the challenges associated with integrating congestion control mechanisms, which may degrade CAP quality in congested networks by reducing transmission efficiency, increasing communication delays. MDRL-CR-DCC, an advanced version of MDRL-CR, incorporates a dynamic adaptation mechanism that responds to fluctuating congestion levels in the network during the selection and transmission of CPMs by CAVs and RSUs. This adaptation is based on leveraging asynchronous learning principles to align with the operational mode of DCC. The results demonstrated that selecting the appropriate relevant objects to adjust the size of CPMs to the size threshold imposed by DCC effectively reduced congestion in the communication channel. This, in turn, improved network reliability and consequently enhanced the CAP resiliency in the varying environment and network conditions imposed by DCC restrictions.

## **7.2 Limitations and Perspectives**

The contributions made by this thesis pave the way for further research and enhancements aimed at bolstering the resilience of CAP in C-ITS. In this section, we outline both short-term goals currently being pursued and long-term perspectives for post-thesis exploration.

### **7.2.1 Short-term Goals**

#### **7.2.1.1 Leveraging DNN Models for Enhanced Modeling**

Object inclusion in the CPM relies on various perception contexts, such as distance from receiving CAVs and potential road occlusions. While our current approach computes object relevance based on a joint function of these factors, it may fall short in providing accurate and realistic values, particularly when faced with complex scenarios. Leveraging DNN models offers a promising avenue to predict more precise relevance values by analyzing real-world datasets, including factors like sensor limitations in adverse weather conditions and intricate road geometry.

#### **7.2.1.2 Consideration of DCC Power and Rate Controls during CPM Transmission**

The optimization problem of MDRL-CR-DCC currently adapts object inclusion and CPM transmission to the message size suggested by the DCC. However, considering only the limited message during transmission may not yield optimal outcomes, as transmit power and rate vary dynamically based on network congestion levels. This dynamic nature can lead to some CAVs receiving outdated information or missing critical updates due to decreased transmission power or generation frequency by the DCC. Integrating signal propagation models and generation

frequency algorithms into the modeling could enhance CAP quality and allows for increased environmental awareness under DCC restrictions.

### 7.2.1.3 Performance Evaluation Based on ETSI's New Specification

Recently, ETSI defined a new specification for the CP service (ETSI TS 103 324 V2.1.1 (2023-06)), introducing a revised CPM format and generation rules to better address network challenges and enhance road safety. The process of object inclusion and transmission of CPMs is primarily based on the usefulness of information, which is calculated using simplistic and static assumptions. Furthermore, the new CPM format facilitates operation using MCO, enabling CAVs and RSUs to transmit relevant objects along with their associated regions within the same CPM. Our ongoing work evaluates the scalability of our contributions in light of this new specification, aiming to ensure their continued effectiveness in enhancing CAP resiliency in C-ITS.

## 7.2.2 Long-term Perspectives

Improving CAP resiliency in C-ITS necessitates addressing broader challenges beyond communication strategies in congested networks and intricate driving environments. Our long-term perspectives encompass:

### 7.2.2.1 Perception Data Quality and Reliability

Addressing challenges related to sensor malfunctions and adverse weather conditions is crucial. Integrating AI techniques to fuse heterogeneous information from various sources will enhance the accuracy and reliability of perception data.

### 7.2.2.2 Cybersecurity

Ensuring resiliency against cybersecurity attacks is paramount. Developing anomaly detection and behavioral analysis techniques to identify and mitigate potential threats will ensure the continuous and safe operation of CAP in C-ITS.

# References

- [1] FESTAG, Andreas. Cooperative intelligent transport systems standards in Europe. *IEEE communications magazine*, 2014, vol. 52, no 12, p. 166-172.
- [2] JIANG, Daniel, CHEN, Qi, et DELGROSSI, Luca. Optimal data rate selection for vehicle safety communications. In: *Proceedings of the fifth ACM international workshop on Vehicular Inter-Networking*. 2008. p. 30-38.
- [3] ETSI TS 102 894-1 V1.1.1 (2013-08) Intelligent Transport Systems (ITS); Users and applications requirements; Part 1: Facility layer structure, functional requirements and specifications
- [4] HAMMI, Badis, MONTEUUIS, Jean-Philippe, et PETIT, Jonathan. PKIs in C-ITS: Security functions, architectures and projects: A survey. *Vehicular Communications*, 2022, vol. 38, p. 100531.
- [5] DEMBA, Albert et MÖLLER, Dietmar PF. Vehicle-to-vehicle communication technology. In: *2018 IEEE international conference on electro/information technology (EIT)*. IEEE, 2018. p. 0459-0464.
- [6] ABBASI, Irshad Ahmed et SHAHID KHAN, Adnan. A review of vehicle to vehicle communication protocols for VANETs in the urban environment. *future internet*, 2018, vol. 10, no 2, p. 14.
- [7] ETSI EN 303 797 V2.1.1 (2023-11) Intelligent Transport Systems (ITS); ITS-G5 Access layer in the 5 GHz frequency band; Release 2
- [8] ETSI EN 302 663 V1.3.1 (2020-01) Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band
- [9] STROM, Erik G. On medium access and physical layer standards for cooperative intelligent transport systems in Europe. *Proceedings of the IEEE*, 2011, vol. 99, no 7, p. 1183-1188.
- [10] ETSI TR 103 439 V2.1.1 (2021-10) Intelligent Transport Systems (ITS); Multi-Channel Operation study; Release 2
- [11] ETSI EN 303 613 V1.1.1 (2020-01) Intelligent Transport Systems (ITS); LTE-V2X Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band
- [12] WANG, Xuyu, MAO, Shiwen, et GONG, Michelle X. An overview of 3GPP cellular vehicle-to-everything standards. *GetMobile: Mobile Computing and Communications*, 2017, vol. 21, no 3, p. 19-25.
- [13] ETSI TS 123 501 V16.6.0 (2020-10) 5G; System architecture for the 5G System (5GS) (3GPP TS 23.501 version 16.6.0 Release 16)
- [14] ADIL, Muhammad, SONG, Houbing, KHAN, Muhammad Khurram, et al. 5G/6G-enabled metaverse technologies: Taxonomy, applications, and open security challenges with future research directions. *Journal of Network and Computer Applications*, 2024, p. 103828.
- [15] ETSI EN 302 636-3 V1.1.2 (2014-03) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 3: Network Architecture
- [16] ETSI TS 102 636-4-2 V1.2.1 (2020-04) 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
- [17] ETSI TS 102 636-6-1 V1.1.1 (2011-03) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 6: Internet Integration; Sub-part 1: Transmission of IPv6 Packets over GeoNetworking Protocols

- [18]V ETSI EN 302 636-5-1 V2.2.1 (2019-05) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol
- [19] ETSI TS 102 723-2 V1.1.1 (2012-11) Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 2: Management information base
- [20] ETSI TS 102 723-3 V1.1.1 (2012-11) Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 3: Interface between management entity and access layer
- [21] ETSI TS 102 723-4 V1.1.1 (2012-11) Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 4: Interface between management entity and networking & transport layer
- [22] ETSI TS 102 723-5 V1.1.1 (2012-11) Intelligent Transport Systems (ITS); OSI cross-layer topics; Part 5: Interface between management entity and facilities layer
- [23] ETSI TR 102 893 V1.2.1 (2017-03) Intelligent Transport Systems (ITS); Security; Threat, Vulnerability and Risk Analysis (TVRA)
- [24] ETSI TS 103 097 V2.1.1 (2021-10) Intelligent Transport Systems (ITS); Security; Security header and certificate formats; Release 2
- [25] HAMMI, Badis, MONTEUUIS, Jean-Philippe, et PETIT, Jonathan. PKIs in C-ITS: Security functions, architectures and projects: A survey. *Vehicular Communications*, 2022, vol. 38, p. 100531.
- [26] YOSHIZAWA, Takahito, SINGELÉE, Dave, MUEHLBERG, Jan Tobias, et al. A survey of security and privacy issues in v2x communication systems. *ACM Computing Surveys*, 2023, vol. 55, no 9, p. 1-36.
- [27] WANG, Zhendong, WEI, Haoran, WANG, Jianda, et al. Security Issues and Solutions for Connected and Autonomous Vehicles in a Sustainable City: A Survey. *Sustainability*, 2022, vol. 14, no 19, p. 12409.
- [28] ETSI TS 102 894-2 V1.3.1 (2018-08) Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary
- [29] ETSI EN 302 637-2 V1.3.2 (2014-11) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service
- [30] ETSI TS 103 900 V2.1.1 (2023-11) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Cooperative Awareness Service; Release 2
- [31] ETSI TS 102 724 (V1.1.1): "Intelligent Transport Systems (ITS); Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band"
- [32] ETSI TS 103 574 V1.1.1 (2018-11) Intelligent Transport Systems (ITS); Congestion Control Mechanisms for the C-V2X PC5 interface; Access layer part
- [33] ETSI TS 103 831 V2.1.1 (2022-11) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Decentralized Environmental Notification Service; Release 2
- [34] ETSI TS 103 300-3 V2.2.1 (2023-02) Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 3: Specification of VRU awareness basic service; Release 2
- [35] ETSI TS 103 300-2 V2.2.1 (2021-04) Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 2: Functional Architecture and Requirements definition; Release 2
- [36] ETSI TR 103 300-1 V2.3.1 (2022-11) Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 1: Use Cases definition; Release 2
- [37] ETSI TR 103 562 V2.1.1 (2019-12) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2

- [38] ETSI TS 103 301 V1.3.1 (2020-02) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services
- [39] ETSI TR 101 612 V1.1.1 (2014-09) Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium; Report on Cross layer DCC algorithms and performance evaluation
- [40] ETSI TS 102 687 V1.2.1 (2018-04) Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part
- [41] KAROUI, Mouna. Congestion mitigation and network selection management in a heterogeneous C-ITS communication architecture. 2021. Thèse de doctorat. Université Clermont Auvergne.
- [42] ETSI TS 103 141 V2.1.1 (2021-11) Intelligent Transport Systems (ITS); Facilities layer; Communication congestion control; Release 2.
- [43] DINH, Hoang T., LEE, Chonho, NIYATO, Dusit, et al. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, 2013, vol. 13, no 18, p. 1587-1611.
- [44] BONOMI, Flavio, MILITO, Rodolfo, ZHU, Jiang, et al. Fog computing and its role in the internet of things. In : *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. 2012. p. 13-16.
- [45] SATYANARAYANAN, Mahadev. The emergence of edge computing. *Computer*, 2017, vol. 50, no 1, p. 30-39.
- [46] HAYDARI, Ammar et YILMAZ, Yasin. Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 2020, vol. 23, no 1, p. 11-32.
- [47] Taxonomy SAE. Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. In: *SAE Standard J3016 (2018)*.
- [48] ARNOLD, Eduardo, AL-JARRAH, Omar Y., DIANATI, Mehrdad, et al. A survey on 3d object detection methods for autonomous driving applications. *IEEE Transactions on Intelligent Transportation Systems*, 2019, vol. 20, no 10, p. 3782-3795.
- [49] LI, You et IBANEZ-GUZMAN, Javier. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Processing Magazine*, 2020, vol. 37, no 4, p. 50-61.
- [50] LU, Tao, DING, Xiang, LIU, Haisong, et al. LinK: Linear Kernel for LiDAR-based 3D Perception. In : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023. p. 1105-1115.
- [51] HE, Chenhang, LI, Ruihuang, LI, Shuai, et al. Voxel set transformer: A set-to-set approach to 3d object detection from point clouds. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022. p. 8417-8427.
- [52] BAFTIU, Irfan, PAJAZITI, Arbnor, et CHEOK, Ka C. Multi-mode surround view for ADAS vehicles. In : *2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*. IEEE, 2016. p. 190-193.
- [53] DELAVARIAN, Mohadeseh, REZA MAROUZI, Omid, HASSANPOUR, Hamid, et al. Multi-camera multiple vehicle tracking in urban intersections based on multilayer graphs. *IET Intelligent Transport Systems*, 2020, vol. 14, no 12, p. 1673-1690.
- [54] Hughes, W.J. Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Analysis Report; FAA GPS Performance Analysis Report, Tech. Rep. 94; The University of Texas at Austin: Austin, TX, USA, July 2016.

- [55] ZHU, Ni, MARAIS, Juliette, BÉTAILLE, David, et al. GNSS position integrity in urban environments: A review of literature. *IEEE Transactions on Intelligent Transportation Systems*, 2018, vol. 19, no 9, p. 2762-2778.
- [56] AHMAD, Khairol Amali Bin, SAHMOUDI, Mohamed, et MACABIAU, Christophe. Characterization of GNSS receiver position errors for user integrity monitoring in urban environments. In : *ENC-GNSS 2014, European Navigation Conference*. 2014.
- [57] WANG, Zhangjing, WU, Yu, et NIU, Qingqing. Multi-sensor fusion in automated driving: A survey. *IEEE Access*, 2019, vol. 8, p. 2847-2868.
- [58] ZHAO, Kun, MA, Lingfei, MENG, Yu, et al. 3D vehicle detection using multi-level fusion from point clouds and images. *IEEE Transactions on Intelligent Transportation Systems*, 2022, vol. 23, no 9, p. 15146-15154.
- [59] YE, Egon, SPIEGEL, Philip, et ALTHOFF, Matthias. Cooperative raw sensor data fusion for ground truth generation in autonomous driving. In : *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020. p. 1-7.
- [60] CHEN, Qi, TANG, Sihai, YANG, Qing, et al. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. In: *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019. p. 514-524.
- [61] ZHOU, Taohua, CHEN, Junjie, SHI, Yining, et al. Bridging the view disparity between radar and camera features for multi-modal fusion 3d object detection. *IEEE Transactions on Intelligent Vehicles*, 2023, vol. 8, no 2, p. 1523-1535.
- [62] V ZHENG, Lianqing, LI, Sen, TAN, Bin, et al. RCFusion: Fusing 4D Radar and Camera with Bird's-Eye View Features for 3D Object Detection. *IEEE Transactions on Instrumentation and Measurement*, 2023.
- [63] ARNOLD, Eduardo, DIANATI, Mehrdad, DE TEMPLE, Robert, et al. Cooperative perception for 3D object detection in driving scenarios using infrastructure sensors. *IEEE Transactions on Intelligent Transportation Systems*, 2020, vol. 23, no 3, p. 1852-1864.
- [64] REN, Kui, WANG, Qian, WANG, Cong, et al. The security of autonomous driving: Threats, defenses, and future directions. *Proceedings of the IEEE*, 2019, vol. 108, no 2, p. 357-372.
- [65] LI, Lantao et SUN, Chen. NLOS Dies Twice: Challenges and Solutions of V2X for Cooperative Perception. *arXiv preprint arXiv:2307.06615*, 2023.
- [66] NGO, Hieu, FANG, Hua, et WANG, Honggang. Cooperative Perception with V2V Communication for Autonomous Vehicles. *IEEE Transactions on Vehicular Technology*, 2023.
- [67] HE, Yuankai, WU, Baofu, DONG, Zheng, et al. Towards C-V2X Enabled Collaborative Autonomous Driving. *IEEE Transactions on Vehicular Technology*, 2023.
- [68] KUMAR, Swarun, SHI, Lixin, AHMED, Nabeel, et al. Carspeak: a content-centric network for autonomous driving. *ACM SIGCOMM Computer Communication Review*, 2012, vol. 42, no 4, p. 259-270.
- [69] KIM, Seong-Woo, QIN, Baoxing, CHONG, Zhuang Jie, et al. Multivehicle cooperative driving using cooperative perception: Design and experimental validation. *IEEE Transactions on Intelligent Transportation Systems*, 2014, vol. 16, no 2, p. 663-680.
- [70] GÜNTHER, Hendrik-Jörn, MENNENGA, Björn, TRAUER, Oliver, et al. Realizing collective perception in a vehicle. In: *2016 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2016. p. 1-8.
- [71] SAE, "V2x sensor-sharing for cooperative & automated driving," *SAE J3224*, 2019.
- [72] FENG, Yijia, HE, Dazhi, NIU, Lei, et al. The overview of Chinese cooperative intelligent transportation system vehicular communication application layer specification and data exchange standard. In: *Information Technology and Intelligent Transportation Systems*. IOS Press, 2017. p. 516-526.



- [73] C. S. of Automotive Engineers, "Cooperative intelligent transportation system vehicular communication application layer specification and data exchange standard (phase 2)," CSAE 157, 2020.
- [74] ANSARI, Mohammad Raashid, MONTEUUIS, Jean-Philippe, PETIT, Jonathan, et al. V2x misbehavior and collective perception service: Considerations for standardization. In: 2021 IEEE Conference on Standards for Communications and Networking (CSCN). IEEE, 2021. p. 1-6.
- [75] RIEBL, Raphael, GÜNTHER, Hendrik-Jörn, FACCHI, Christian, et al. Artery: Extending veins for VANET applications. In: 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS). IEEE, 2015. p. 450-456.
- [76] KRAJZEWICZ, Daniel, ERDMANN, Jakob, BEHRISCH, Michael, et al. Recent development and applications of SUMO-Simulation of Urban MObility. International journal on advances in systems and measurements, 2012, vol. 5, no 3&4.
- [77] SOMMER, Christoph, ECKHOFF, David, BRUMMER, Alexander, et al. Veins: The open source vehicular network simulation framework. Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem, 2019, p. 215-252.
- [78] GUNTHER, Hendrik-jorn, TRAUER, Oliver, et WOLF, Lars. The potential of collective perception in vehicular ad-hoc networks. In: 2015 14th International Conference on ITS Telecommunications (ITST). IEEE, 2015. p. 1-5.
- [79] ALLIG, Christoph et WANIELIK, Gerd. Extending the vehicular network simulator Artery in order to generate synthetic data for collective perception. Advances in Radio Science, 2019, vol. 17, p. 189-196.
- [80] BAI, Zhengwei, WU, Guoyuan, QI, Xuewei, et al. Infrastructure-based object detection and tracking for cooperative driving automation: A survey. In: 2022 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2022. p. 1366-1373.
- [81] BAI, Zhengwei, WU, Guoyuan, BARTH, Matthew J., et al. Pillargrid: Deep learning-based cooperative perception for 3d object detection from onboard-roadside lidar. In: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2022. p. 1743-1749.
- [82] MENG, Zonglin, XIA, Xin, XU, Runsheng, et al. HYDRO-3D: Hybrid Object Detection and Tracking for Cooperative Perception Using 3D LiDAR. IEEE Transactions on Intelligent Vehicles, 2023.
- [83] MOUAWAD, Nadia et MANNONI, Valérian. Collective perception messages: New low complexity fusion and v2x connectivity analysis. In: 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall). IEEE, 2021. p. 1-5.
- [84] SHANGGUAN, Wei, DU, Yu, et CHAI, Linguo. Interactive perception-based multiple object tracking via CVIS and AV. IEEE Access, 2019, vol. 7, p. 121907-121921.
- [85] ASVADI, Alireza, GIRAO, Pedro, PEIXOTO, Paulo, et al. 3D object tracking using RGB and LIDAR data. In : 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). IEEE, 2016. p. 1255-1260.
- [86] CHEN, Qi, MA, Xu, TANG, Sihai, et al. F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds. In: Proceedings of the 4th ACM/IEEE Symposium on Edge Computing. 2019. p. 88-100.
- [87] WANG, Yicong, DE VECIANA, Gustavo, SHIMIZU, Takayuki, et al. Performance and scaling of collaborative sensing and networking for automated driving applications. In: 2018 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, 2018. p. 1-6.
- [88] LI, Tao, HAN, Xu, et MA, Jiaqi. Cooperative perception for estimating and predicting microscopic traffic states to manage connected and automated traffic. IEEE Transactions on Intelligent Transportation Systems, 2021, vol. 23, no 8, p. 13694-13707.

- [89] THANDAVARAYAN, Gokulnath, SEPULCRE, Miguel, et GOZALVEZ, Javier. Analysis of message generation rules for collective perception in connected and automated driving. In: 2019 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2019. p. 134-139.
- [90] THANDAVARAYAN, Gokulnath, SEPULCRE, Miguel, et GOZALVEZ, Javier. Redundancy mitigation in cooperative perception for connected and automated vehicles. In: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). IEEE, 2020. p. 1-5.
- [91] HIGUCHI, Takamasa, GIORDANI, Marco, ZANELLA, Andrea, et al. Value-anticipating v2v communications for cooperative perception. In: 2019 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2019. p. 1947-1952.
- [92] DELOOZ, Quentin, WILLECKE, Alexander, GARLICH, Keno, et al. Analysis and evaluation of information redundancy mitigation for V2X collective perception. *IEEE Access*, 2022, vol. 10, p. 47076-47093.
- [93] CHTOUROU, Ameni, MERDRIGNAC, Pierre, et SHAGDAR, Oyunchimeg. Context-aware content selection and message generation for collective perception services. *Electronics*, 2021, vol. 10, no 20, p. 2509.
- [94] HUANG, Hui, FANG, Wenqi, et LI, Huiyun. Performance modelling of v2v based collective perceptions in connected and autonomous vehicles. In: 2019 IEEE 44th Conference on Local Computer Networks (LCN). IEEE, 2019. p. 356-363.
- [95] H. Huang et al., Data Redundancy Mitigation in V2X Based Collective Perceptions, *IEEE Access*, vol. 8, pp. 13405-13418, Jan. 2020. <https://doi.org/10.1109/ACCESS.2020.2965552>
- [96] XU, Haowei et LIU, Xiping. Perception synergy optimization with deep reinforcement learning for cooperative perception in C-V2V scenarios. *Vehicular Communications*, 2022, vol. 38, p. 100536.
- [97] AOKI, Shunsuke, HIGUCHI, Takamasa, et ALTINTAS, Onur. Cooperative perception with deep reinforcement learning for connected vehicles. In: 2020 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2020. p. 328-334.
- [98] ABDEL-AZIZ, Mohamed K., PERFECTO, Cristina, SAMARAKOON, Sumudu, et al. Vehicular cooperative perception through action branching and federated reinforcement learning. *IEEE Transactions on Communications*, 2021, vol. 70, no 2, p. 891-903.
- [99] JUNG, Beopgwon, KIM, Joonwoo, et PACK, Sangheon. Deep Reinforcement Learning-based Context-Aware Redundancy Mitigation for Vehicular Collective Perception Services. In: 2022 International Conference on Information Networking (ICOIN). IEEE, 2022. p. 276-279.
- [100] ARULKUMARAN, Kai, DEISENROTH, Marc Peter, BRUNDAGE, Miles, et al. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 2017, vol. 34, no 6, p. 26-38.
- [101] DULAC-ARNOLD, Gabriel, EVANS, Richard, VAN HASSELT, Hado, et al. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.
- [102] TSITSIKLIS, John et VAN ROY, Benjamin. Analysis of temporal-difference learning with function approximation. *Advances in neural information processing systems*, 1996, vol. 9.
- [103] VAN HASSELT, Hado, GUEZ, Arthur, et SILVER, David. Deep reinforcement learning with double q-learning. In : *Proceedings of the AAAI conference on artificial intelligence*. 2016.
- [104] HAJI, Saad Hikmat et ABDULAZEEZ, Adnan Mohsin. Comparison of optimization techniques based on gradient descent algorithm: A review. *PalArch's Journal of Archaeology of Egypt/Egyptology*, 2021, vol. 18, no 4, p. 2715-2743.
- [105] BAZZI, Alessandro, SEPULCRE, Miguel, DELOOZ, Quentin, et al. Multi-Channel Operation for the Release 2 of ETSI Cooperative Intelligent Transport Systems. *arXiv preprint arXiv:2305.19863*, 2023.

- [106] ZHOU, Pengyuan, KORTOÇI, Pranvera, YAU, Yui-Pan, et al. AICP: Augmented informative cooperative perception. *IEEE Transactions on Intelligent Transportation Systems*, 2022, vol. 23, no 11, p. 22505-22518.
- [107] BAI, Zhengwei, WU, Guoyuan, BARTH, Matthew J., et al. Cooperverse: A Mobile-Edge-Cloud Framework for Universal Cooperative Perception with Mixed Connectivity and Automation. *arXiv preprint arXiv:2302.03128*, 2023.
- [108] THANDAVARAYAN, Gokulnath, SEPULCRE, Miguel, GOZALVEZ, Javier, et al. Scalable cooperative perception for connected and automated driving. *Journal of Network and Computer Applications*, 2023, vol. 216, p. 103655.
- [109] YUAN, Yunshuang, CHENG, Hao, et SESTER, Monika. Keypoints-based deep feature fusion for cooperative vehicle detection of autonomous driving. *IEEE Robotics and Automation Letters*, 2022, vol. 7, no 2, p. 3054-3061.
- [110] WANG, Tianhang, CHEN, Guang, CHEN, Kai, et al. UMC: A Unified Bandwidth-efficient and Multi-resolution based Collaborative Perception Framework. *arXiv preprint arXiv:2303.12400*, 2023.
- [111] YUAN, Yunshuang, CHENG, Hao, YANG, Michael Ying, et al. Generating Evidential BEV Maps in Continuous Driving Space. *arXiv preprint arXiv:2302.02928*, 2023.
- [112] LI, H., SIMA, C., DAI, J., et al. Delving into the Devils of Bird's-eye-view Perception: A Review. *Evaluation and Recipe*. *arXiv*, 2022.x
- [113] JOSANG, Audun et HALLER, Jochen. Dirichlet reputation systems. In: *The Second International Conference on Availability, Reliability and Security (ARES'07)*. IEEE, 2007. p. 112-119.
- [114] JIA, Yukuan, MAO, Ruiqing, SUN, Yuxuan, et al. Online v2x scheduling for raw-level cooperative perception. In: *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022. p. 309-314.
- [115] ALMEIDA, Carolina P., GONÇALVES, Richard A., VENSKE, Sandra, et al. Hyper-heuristics using multi-armed bandit models for multi-objective optimization. *Applied Soft Computing*, 2020, vol. 95, p. 106520.
- [116] JIA, Yukuan, MAO, Ruiqing, SUN, Yuxuan, et al. MASS: Mobility-Aware Sensor Scheduling of Cooperative Perception for Connected Automated Driving. *IEEE Transactions on Vehicular Technology*, 2023.
- [117] HU, Yue, FANG, Shaoheng, LEI, Zixing, et al. Where2comm: Communication-efficient collaborative perception via spatial confidence maps. *Advances in neural information processing systems*, 2022, vol. 35, p. 4874-4886.
- [118] HUANG, Tao, LIU, Jianan, ZHOU, Xi, et al. V2X cooperative perception for autonomous driving: Recent advances and challenges. *arXiv preprint arXiv:2310.03525*, 2023.
- [119] HAN, Yushan, ZHANG, Hui, LI, Huifang, et al. Collaborative perception in autonomous driving: Methods, datasets and challenges. *arXiv preprint arXiv:2301.06262*, 2023.
- [120] ELGHAZALY, Gamal, FRANK, Raphaël, HARVEY, Scott, et al. High-definition maps: Comprehensive survey, challenges and future perspectives. *IEEE Open Journal of Intelligent Transportation Systems*, 2023.
- [121] REKKAS, Vasileios P., SOTIROUDIS, Sotirios, SARIGIANNIDIS, Panagiotis, et al. Machine learning in beyond 5G/6G networks—State-of-the-art and future trends. *Electronics*, 2021, vol. 10, no 22, p. 2786.
- [122] KONDA, Vijay et TSITSIKLIS, John. Actor-critic algorithms. *Advances in neural information processing systems*, 1999, vol. 12.
- [123] TSITSIKLIS, John et VAN ROY, Benjamin. Analysis of temporal-difference learning with function approximation. *Advances in neural information processing systems*, 1996, vol. 9.

- [124] MNIH, Volodymyr, BADIA, Adria Puigdomenech, MIRZA, Mehdi, et al. Asynchronous methods for deep reinforcement learning. In: International conference on machine learning. PMLR, 2016. p. 1928-1937.
- [125] LOWE, Ryan, WU, Yi I., TAMAR, Aviv, et al. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 2017, vol. 30.
- [126] SUTTON, Richard S., MCALLESTER, David, SINGH, Satinder, et al. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 1999, vol. 12.
- [127] THANDEVARAYAN, Gokulnath, SEPULCRE, Miguel, et GOZALVEZ, Javier. Generation of cooperative perception messages for connected and automated vehicles. *IEEE Transactions on Vehicular Technology*, 2020, vol. 69, no 12, p. 16336-16341.
- [128] BALADOR, Ali, CINQUE, Elena, PRATESI, Marco, et al. Survey on decentralized congestion control methods for vehicular communication. *Vehicular Communications*, 2022, vol. 33, p. 100394.
- [129] XU, Q., PAN, J. A., LI, K. Q., et al. Design of connected vehicle controller under cloud control scenes with unreliable communication. *Automot. Eng*, 2021, vol. 43, p. 527-536.
- [130] XUE-YANG, CHANG, QING, X. U., KE-QIANG, L. I., et al. Analysis of intelligent and connected vehicle control under communication delay and packet loss. *China Journal of Highway and Transport*, 2019, vol. 32, no 6, p. 216.
- [131] GÜNTHER, Hendrik-Jörn, RIEBL, Raphael, WOLF, Lars, et al. The effect of decentralized congestion control on collective perception in dense traffic scenarios. *Computer Communications*, 2018, vol. 122, p. 76-83.
- [132] DELOOZ, Quentin et FESTAG, Andreas. Network load adaptation for collective perception in V2X communications. In: 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE). IEEE, 2019. p. 1-6.
- [133] GÜNTHER, Hendrik-Jorn, RIEBL, Raphael, WOLF, Lars, et al. Collective perception and decentralized congestion control in vehicular ad-hoc networks. In: 2016 IEEE Vehicular Networking Conference (VNC). IEEE, 2016. p. 1-8.
- [134] SEWAK, Mohit et SEWAK, Mohit. Actor-Critic Models and the A3C: The Asynchronous Advantage Actor-Critic Model. *Deep reinforcement learning: frontiers of artificial intelligence*, 2019, p. 141-152.
- [135] PACZOLAY, Gabor et HARMATI, Istvan. A New Advantage Actor-Critic Algorithm For Multi-Agent Environments. In: 2020 23rd International Symposium on Measurement and Control in Robotics (ISMCR). IEEE, 2020. p. 1-6.