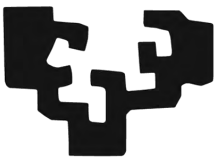
eman ta zabal zazu



Universidad del País Vasco
Euskal Herriko Unibertsitatea

University of the Basque Country
Electicity and Electronic

COOPERATIVE MANEUVERS FOR URBAN ENVIRONMENTS WITH AUTOMATED VEHICLES

Carlos Eduardo Hidalgo Vieira

April, 2024

COOPERATIVE MANEUVERS FOR URBAN ENVIRONMENTS WITH AUTOMATED VEHICLES

Carlos Eduardo Hidalgo Vieira

Thesis Advisors:

Dr. Estibaliz Asua Uriarte, University of the Basque Country.

Dr. Joshué Manuel Pérez Rastelli, Tecnalia Member of Basque Research & Technology Alliance.

To Myriam, Emilia and the little ones, I hope will come in the future.

ACKNOWLEDGEMENTS

Sin duda alguna, esta parte del libro ha sido la más compleja de escribir. Ha sido una larga aventura llena de altos, bajos y personas increíbles que hicieron de este viaje algo hermoso y gratificante. Por eso, plasmar todos mis agradecimientos en un espacio tan limitado me resulta muy complicado y más en inglés, asi que ahí va en español jajajajaja.

Quiero comenzar expresando mi profundo agradecimiento a mis tutores, Joshué y Estibalitz. Sin ustedes, esta aventura ni siquiera habría comenzado. Agradezco su confianza, orientación y la oportunidad que me brindaron hace 5 años y medio, justo después de salir de la universidad. En especial, a ti Joshué, que a pesar de una presentación desastrosa en las prácticas, decidiste confiar en mí y ofrecerme la posibilidad de embarcarme en este reto tan especial.

A continuación, quiero agradecer a mi familia, ya que, de igual forma que sin mis tutores, sin ellos esta aventura jamás hubiera comenzado. Mis padres, Rubén Darío y María Emilia, mis hermanos Andrea Carolina y Rubén Alejandro, mis cuñados Helio y Alejandra, y por supuesto, mis sobrinos, Ricardo e Isabella. No solo con sus constantes "¿Y pa cuándo la tesis?", me apoyaron y ayudaron, sino que incluso desde la distancia, fueron fundamentales para que pudiera llegar hasta aquí y convertirme en la persona que soy hoy en día.

A nivel personal, a mis amigos, los "Electrovagos", los cuales jugaron un papel muy importante durante este proceso. Julio, Walter, Tomás, Nievsabel, Andrea, Gabriel, María Gabriela y Víctor. Muchas gracias por estar ahí. Por escucharme, animarme y darme fuerzas en los momentos de frustración y dificultades. Su apoyo incondicional me motivó para seguir adelante cuando pensaba que no podía más.

En el ámbito laboral, quiero agradecer a mis compañeros del grupo CCAM. Leonardo, Ray, José Ángel, Mauricio, Sergio, Jesús, Iker, Fernando, Joseba, Alberto, Javier, y por supuesto! al "team Go Mobility", Asier y Mario, cada uno de ustedes contribuyó de manera directa o indirecta a mi crecimiento profesional y al desarrollo de esta tesis. Su ayuda fue invaluable y les estaré eternamente agradecido.

Y por último, pero no menos importante, quiero expresar mi profundo agradecimiento a la persona que entra en todas estas categorías y aún más: mi familia, mi amiga, mi compañera de trabajo y mi confidente. A ti, mi futura esposa, Myriam. Al comienzo de esta tesis, nunca imaginé que te convertirías en la persona más importante e influyente en mi vida. A pesar de todos los dolores de cabeza que me das, gracias por todo tu apoyo, paciencia y amor incondicional. Sin ti, definitivamente

no habría podido superar todos los desafíos. En cada aspecto de mi vida, tú estás ahí para ayudarme, apoyarme, animarme y, sobre todo, aguantarme jajajajajaja. Además, me has dado el regalo más grande que podría pedir, nuestra hija Emilia, quien desde hace un año ha puesto nuestas vidas patas arriba a cambio de llenarla del amor más puro y de felicidad. Una vez más, gracias por todo.

COOPERATIVE MANEUVERS FOR URBAN ENVIRONMENTS WITH AUTOMATED VEHICLES

Abstract

by Carlos Eduardo Hidalgo Vieira
Universidad del País Vasco - Euskal Erriko Unibertsitatea
2024

Thesis Advisors:
Estibalitz Asua Uriarte, Ph.D.
Joshué Manuel Pérez Rastelli, Ph.D.

Road safety remains a critical global concern. Human-driven motor vehicles contribute significantly to the high number of accidents, highlighting the urgent need for effective solutions. Traffic congestion intensifies road safety issues, with studies showing a direct correlation between congestion and accident frequency. Despite improvements in road infrastructure, challenges persist, particularly in navigating complex road segments like roundabouts and intersections. Cooperative, Connected, and Automated Mobility (CCAM) technologies emerge as a promising solution to enhance transportation efficiency and safety. CCAM aims to optimize traffic flow, mitigate congestion, and improve safety through the integration of Connected Automated Vehicles (CAVs) with advanced infrastructure technologies and algorithms.

Cooperative maneuvers play a pivotal role in the realization of CCAM technologies, offering a strategic approach to address challenges associated with complex road segments. By using advanced communication systems, CAVs can exchange real-time data about their positions, speed, and intentions, allowing for coordination and decision-making during maneuvers. Moreover, cooperative maneuvers not only enhance traffic safety but also optimize transport flow, thereby reducing congestion and improving overall transportation efficiency. As such, the development and implementation of cooperative maneuvers represent a crucial step towards achieving the overarching goals of CCAM technologies in revolutionizing the future of transportation.

Furthermore, as these technologies advance, ensuring robust cyber-security measures becomes imperative to safeguard against potential cyber-threats. The integration of CAVs with sophisticated communication systems exposes them to vulnerabilities, making them susceptible to cyber-attacks that could compromise

vehicle control and endanger road safety. By fortifying cyber-security frameworks, initiatives aim to bolster the protection of CAVs and their communication networks, thereby fostering trust and reliability in the deployment of cooperative maneuvers. As CCAM continues to evolve, cyber-security remains a critical aspect that requires ongoing attention and investment to uphold the integrity and safety of future transportation systems.

Aligned with the aforementioned premises, this Ph.D. thesis aims to address the coordination of multiple CAVs to execute various cooperative maneuvers across diverse testing environments, while also delving briefly into cyber-security frameworks based on Internet of Things (IoT) solutions. To fulfill this objective, the thesis embarks with a comprehensive review of the State-of-the-Art (SoA), focusing on the advancements in driving architecture constituting the CCAM systems, alongside the current state of cooperation in its domain. Following the identification of challenges pertinent, the thesis introduces the Automated Driving Core (AUDRIC) architecture, serving as the foundational framework. Particular emphasis is placed on integrating the SerIoT and IoTAC systems into the infrastructure module.

Subsequently, the thesis delineates the validation platforms, encompassing both simulated and real environments, along with the various proving grounds where these tests were conducted. It then elucidates the decision and control algorithms developed for executing cooperative maneuvers. The primary algorithm pertains to car following, comprising Adaptive Cruise Control (ACC) and Cooperative Adaptive Cruise Control (CACC) technologies, supported by a feedback/feedforward + Proportional Derivative (PD) controller framework. Additionally, two decision strategies are elaborated upon: Hybrid Trajectory Planning (HYTP), using Bézier Curves and predictive control, and Real-Time Trajectory Planning (RTTP), employing high-definition maps and a Finite State Machine (FSM) capable of managing cooperative and non-cooperative scenarios, thus facilitating real-time trajectory planning.

The thesis proceeds to validate cooperative maneuvers, including car following, roundabout merging, platoon lane merging, fleet management, and intersection management, across a diverse array of test environments utilizing the developed algorithms. Based on the results obtained, the thesis concludes by emphasizing the efficacy of the planned strategies and advocating for the expansion of cooperative maneuvers, given their pivotal role in achieving optimal connected and automated driving.

RESUMEN

En la actualidad, la seguridad vial continúa siendo una preocupación global de suma importancia, siendo los vehículos conducidos por humanos una contribución significativa al elevado número de accidentes. La congestión del tráfico agrava dichos problemas, con estudios que muestran una correlación directa entre la congestión y la frecuencia de accidentes. A pesar de las mejoras en la infraestructura vial, persisten desafíos, especialmente en la conducción de tramos de carretera como las rotondas y los cruces. Las tecnologías de Movilidad Cooperativa, Conectada y Automatizada (CCAM, por sus siglas en inglés) emergen como una solución prometedora para mejorar la eficiencia y la seguridad del transporte. CCAM tiene como objetivo optimizar el flujo de tráfico, mitigar la congestión y mejorar la seguridad mediante la integración de Vehículos Automatizados y Conectados (CAV, por sus siglas en inglés) con tecnologías de infraestructura avanzadas y algoritmos.

Las maniobras cooperativas son fundamentales para la implementación de las tecnologías CCAM, ya que ofrecen una solución para abordar los desafíos asociados con segmentos de via complejos. Al aprovechar sistemas de comunicación avanzados, los vehículos pueden intercambiar datos en tiempo real sobre sus posiciones, velocidades e intenciones, lo que permite una coordinación y toma de decisiones durante las maniobras. Además, las maniobras cooperativas no solo mejoran la seguridad del tráfico, sino que también optimizan el flujo del mismo, reduciendo así la congestión y mejorando la eficiencia del transporte en general. Como tal, el desarrollo e implementación de maniobras cooperativas representan un paso crucial para lograr los objetivos generales de las tecnologías CCAM en la revolución del futuro del transporte.

Además, a medida que estos desarrollos avanzan, garantizar medidas sólidas de ciberseguridad se vuelve imperativo para protegerse contra posibles amenazas cibernéticas. La integración de CAVs con sistemas de comunicación sofisticados los expone a vulnerabilidades, haciéndolos susceptibles a ciberataques que podrían comprometer el control del vehículo y poner en peligro la seguridad vial. Al fortalecer los marcos de ciberseguridad, las iniciativas tienen como objetivo reforzar la protección de los CAVs y sus redes de comunicación, fomentando así la confianza y la fiabilidad en la implementación de maniobras cooperativas. A medida que CCAM continúa evolucionando, la ciberseguridad sigue siendo un aspecto crítico que

requiere atención e inversiones continuas para mantener la integridad y la seguridad de los futuros sistemas de transporte.

Alineada con las premisas mencionadas, esta tesis de doctorado tiene como objetivo abordar la coordinación de múltiples CAVs para ejecutar maniobras cooperativas en diversos entornos de prueba, y explorar brevemente marcos de ciberseguridad basados en soluciones de Internet de las Cosas (IoT, por sus siglas en inglés).

Para cumplir con este objetivo, la tesis comienza con una revisión exhaustiva del Estado del Arte (SoA, por sus siglas en inglés), centrándose en los avances de la arquitectura de conducción que constituyen los sistemas CCAM, junto con el estado actual de la cooperación de dichas tecnologías. Tras la identificación de los desafíos pertinentes en este ámbito, la tesis introduce la arquitectura Conducción Automatizada (AUDRIC, por sus siglas en inglés), que sirve como marco fundamental para el desarrollo de la misma. Además, se pone énfasis en la integración de los sistemas SerIoT e IoTAC en el módulo de infraestructura.

Posteriormente, la tesis delinea las plataformas de validación, que abarcan tanto entornos simulados como reales, junto con los escenarios de prueba donde se llevaron a cabo los diferentes estudios. Luego, se explican los algoritmos de decisión y control desarrollados para ejecutar las maniobras cooperativas. El primer algoritmo se refiere al seguimiento vehicular, que comprende tecnologías de Control de Crucero Adaptativo (ACC, por sus siglas en inglés) y Control de Crucero Adaptativo Cooperativo (CACC, por sus siglas en inglés), respaldado por un marco de controlador de retroalimentación / avance + Proporcional Derivativo (PD). Además, se detallan dos estrategias de decisión: Planificación de Trayectoria Híbrida (HYTP, por sus siglas en inglés), que utiliza curvas de Bézier y control predictivo, y Planificación de Trayectoria en Tiempo Real (RTTP, por sus siglas en inglés), que emplea mapas de alta definición y una Máquina de Estados Finitos (FSM, por sus siglas en inglés) capaz de gestionar escenarios cooperativos y no cooperativos, facilitando así la planificación de trayectorias en tiempo real.

La tesis procede a validar las maniobras cooperativas, incluido el seguimiento vehicular, la incorporación a rotondas, la incorporación de carriles en formaciones de pelotones, la gestión de flotas y la gestión de intersecciones, en una variedad de entornos de prueba utilizando los algoritmos desarrollados. Basándose en los resultados obtenidos, la tesis concluye enfatizando la eficacia de las estrategias planificadas y abogando por la expansión de las maniobras cooperativas, dada su importancia crucial en el logro de una conducción conectada y automatizada óptima.

Contents

ACKNOWLEDGEMENTS v

ABSTRACT vii

RESUMEN ix

LIST OF FIGURES xiv

LIST OF TABLES xviii

LIST OF ACRONYMS xx

LIST OF SYMBOLS xxiii

1 Introduction 1

1.1 Background and Motivation . . . 3

1.1.1 From Automated Driving to Cooperative, Connected, and Automated Mobility in Europe . . . 4

1.2 Objectives . . . 6

1.3 Manuscript Organization . . . 7

1.4 Contributions . . . 8

1.5 Dissemination . . . 8

1.5.1 Conferences . . . 8

1.5.2 Journals . . . 9

1.5.3 Book Chapter . . . 9

2 State of the Art 10

2.1 Driving Architecture . . . 12

2.1.1 Acquisition . . . 13

2.1.2 Perception . . . 13

2.1.3 Communication . . . 14

2.1.3.1 DSRC/C-ITS . . . 16

2.1.3.2 Cellular 4G/5G . . . 17

2.1.3.3 Cyber-security . . . 18

2.1.4 Decision . . . 18

2.1.4.1 Global Planner . . . 19

2.1.4.2 Behavioral Planner . . . 21

2.1.4.3 Local Planner . . . 23
2.1.5 Control . . . 24
2.1.6 Actuation . . . 26
2.1.7 Human-Machine Interface . . . 27
2.1.8 Infrastructure . . . 27
2.2 Cooperative, Connected, and Automated Mobility . . . 28
2.2.1 Car Following . . . 31
2.2.1.1 Platooning . . . 33
2.2.2 Intersection Management . . . 36
2.2.3 Cooperative Overtaking . . . 38
2.2.4 Cooperative Merging . . . 40
2.2.4.1 Platoon Merging . . . 43
2.3 Summary and Conclusions . . . 43

3 Validation Framework **45**
3.1 AUDRIC: AUtomated DRIving Core . . . 45
3.1.1 Infrastructure with SerIoT framework . . . 48
3.1.2 Infrastructure with IoTAC framework . . . 50
3.2 Experimental set-up . . . 52
3.2.1 Virtual Platforms . . . 52
3.2.1.1 MATLAB/Simulink with Dynacar and Visor 3D . . . 52
3.2.1.2 ROS with Carla . . . 54
3.2.2 Real Platforms . . . 57
3.2.2.1 Renault Twizy 80 . . . 57
3.2.2.2 Irizar Bus . . . 59
3.2.3 Mixed Environment . . . 60
3.2.4 Test Tracks . . . 60
3.3 Summary and Conclusions . . . 62

4 Cooperative Maneuver Decision and Control Developments **64**
4.1 Car Following Strategy . . . 66
4.2 Hybrid Trajectory Planning Method . . . 71
4.2.1 Nominal Trajectory Calculator . . . 72
4.2.1.1 Global and Local Planner . . . 73
4.2.2 Maneuver Planner . . . 76
4.2.3 Speed Planning For Cooperative Maneuvers . . . 79
4.3 Real-Time Trajectory Planning Method . . . 83
4.3.1 RTTP: Global Planner . . . 83
4.3.2 RTTP: Behavioral Planner . . . 85
4.3.3 RTTP: Local Planner . . . 89
4.3.4 Speed Planning For Cooperative Maneuvers . . . 91
4.4 Maneuvers Negotiation . . . 92

4.5 Summary and Conclusions 98

5 Results and Discussions **99**

5.1 Car Following Strategy 100

5.1.1 ACC and CACC Simulated Performance 101

5.1.2 Renault Twizy 80 Real Test Performance 102

5.1.3 Irizar i2eBus Real Test Performance 104

5.1.4 Car Following Strategy Simulated Performance 106

5.1.5 Discussions 107

5.2 Hybrid Trajectory Planning 109

5.2.1 Roundabout Merging Simulated Performance 109

5.2.2 Roundabout Merging Mixed Test Environment Performance . 110

5.2.3 Platoon Lane Merging Simulated Performance 114

5.2.4 Fleet Management and Smart Intersection Maneuvers Under the SerIoT Framework 117

5.2.4.1 SerIoT System Performance 122

5.2.5 Discussions 122

5.3 Real Time Trajectory Planning 123

5.3.1 Platoon Lane Merging Simulated Performance 123

5.3.2 Case 1: Platoon Lane Merging Mixed Environment Performance 128

5.3.3 Case 2: Platoon Lane Merging with the IoTAC Framework . 132

5.3.4 Discussions 137

5.4 Summary and Conclusions 139

6 Conclusions **141**

6.1 Concluding Remarks 141

6.2 Research Perspective and Future Works 143

Bibliography **145**

List of Figures

1.1 Overview of the most relevant EU founded projects that support the CCAM technologies between 2015-2023 [11]. 5

2.1 Representation of SAE automation levels [18]. 11
2.2 Abstract overview of the control architecture [19]. 12
2.3 Examples of different Perception systems, RADAR, LiDAR and camera. 14
2.4 Different communication types that composes the V2X. 15
2.5 Infrastructure examples. 28
2.6 C2C-CC Road-map [74]. 29
2.7 Examples of the platooning in function of the network topology. 35
2.8 Illustration of the different types of intersections, visually demonstrating potential conflicts through the respective trajectories of each vehicle, denoted in green arrows. 36
2.9 Cooperative overtaking example, where two vehicles (illustrated in red) in the right lane intend to pass a third vehicle (illustrated in green). . 39
2.10 Illustration of various types of merging scenarios, with particular emphasis on cooperative merging cases. 40

3.1 AUDRIC driving architecture. The filled blocks represents where the main contributions of this thesis. 47
3.2 Infrastructure scheme with the SerIoT system [188]. 49
3.3 Infrastructure scheme with the IoTAC system. 51
3.4 AUDRIC architecture represented in Simulink. 53
3.5 Dynacar simulator. 54
3.6 Simulation in CARLA. 55
3.7 NVIDIA PhysX vehicle model [1] 56
3.8 Real platforms used. From left to right: A 12-m Irizar i2eBus, an EMT Gulliver and a Renault Twizy 80 Under the EU project SHOW. 57
3.9 Renault Twizy 80 instrumented, with different sensors and devices to fulfill CAV functionalities. 58
3.10 Irizar 12-m i2eBus, with different sensors and antennas installed to achieve CAVs functionalities. 59

[1] **Webpage**: NVIDIA PhysX SDK → *https://docs.nvidia.com/gameworks/content/ gameworkslibrary/physx/guide/Manual/Vehicles.html.*

3.11 Three mixed scenario configurations used during the Ph.D thesis. Each involving the usage of the Renault Twizy 80, with different simulation set-ups. 60
3.12 Tecnalia Test Track view from from Google Earth. 61
3.13 Ficoba Exhibition Track view from Google Earth. 61
3.14 EMT depot view from Google Earth. 62
3.15 Tecnalia surrounding scenario view from Google Earth. 62

4.1 Control structure for ACC and CACC techniques performed in the thesis. 67
4.2 Delay vs minimum time gap analysis. 69
4.3 String stability frequency analysis 70
4.4 HYTP block diagram. Consisting on two main modules; Nominal Trajectory Calculator, and the Maneuver Planner. 72
4.5 Example of a Bézier curve, indicating how the curve is plotted on the basis of the control points. 73
4.6 Simple Map and trajectory planning on lane change, with the corresponding Bézier control points. 74
4.7 Simple Map and trajectory planning on roundabout of one lane, with the corresponding Bézier control points. 75
4.8 Speed and position response comparative when different controller are used.. 80
4.9 Comparative between the Simulink clock and the real time clock with different MPC sample configurations. 80
4.10 Images sequence of the merging at roundabout with vehicles projections, where the green vehicle generates projections of both vehicles in order to enter the roundabout. 81
4.11 Illustration of platoon merging maneuver as sequence of steps, from top to bottom picture. Where the Platoon B seeks to join Platoon A to form Platoon C. 82
4.12 Views of the Tecnalia Test track according the JOSM application and the drivable space. 85
4.13 FSM state diagram, where the states are: Lane Keeping, Car Following, Merging, Overtaking, and Parking. The numbers are the conditions to transition from one state to another (explained in the text). 87
4.14 Bézier control points positioning example, indicating th directions, the distance between points, as well as the distance considerations in the generated trajectory. 89
4.15 Flow state diagram of the platoon driving, from the initial negotiation point until the end of the maneuver. This case is presented when the vehicles are located in the same lane. 96

4.16 Flow state diagram of the platoon merging, from the initial negotiation point until the end of the maneuver. This case is presented when the vehicles are located in different lanes. 97

5.1 Time-line of each result accomplishment in concordance with the respective project. 99
5.2 Images of each test performed to validate the car following strategy. . 101
5.3 Simulated performance of a platoon of 4 vehicles using ACC. 103
5.4 Simulated performance of a 4 vehicle platoon using CACC. 103
5.5 Real car following performance with a platoon of two Renault Twizy. . 104
5.6 Real car following performance with a platoon of one Irizar I2eBus and one Gulliver. 105
5.7 Simulated performance of a 4 vehicle platoon using the car following strategy in an urban environment with different road components. . . 107
5.8 Simulated roundabout merging using the HYTP approach with two vehicles. 110
5.9 Simulated performance of the merging vehicle using the HYTP. 111
5.10 Images sequence of the merging at roundabout. 112
5.11 Mixed environment performance of the roundabout merging, using the HYTP approach. 113
5.12 Simulated platoon lane merging using 5 vehicles with HYTP alongside the CACC. 115
5.13 Platoon A and Platoon B lane merging performance using the HYTP alongside the car following strategy using only CACC. 116
5.14 Platoon C longitudinal performance using the HYTP alongside the car following strategy using only CACC. 117
5.15 Lane change performance of the Platoon B using the HYTP alongside the car following strategy using only CACC. 118
5.16 Fleet Management scenario description within the SerIoT project [188]. 119
5.17 Fleet management maneuver sequence in traffic jam situation. 119
5.18 Smart Intersection scenario description within the SerIoT project [188]. 120
5.19 Smart intersection maneuver sequence. 121
5.20 Images sequence of the simulated platoon lane merging using 6 vehicles with RTTP approach alongside the car following strategy. 125
5.21 Simulated longitudinal performance of Platoon A and B during the platoon lane merging, using the RTTP approach alongside the car following strategy. 126
5.22 Simulated longitudinal performance of the Platoon C during the platoon lane merging, using the real time trajectory approach alongside the car following strategy. 127
5.23 Platoon B simulated lateral performance during platoon lane merging, using the RTTP approach alongside the car following strategy. 128

5.24 Images sequence of the mix test environment platoon lane merging using 3 vehicles with RTTP approach alongside the car following strategy. 129
5.25 Longitudinal performance of Platoon A and the merging vehicle during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy. 130
5.26 Platoon B mixed test environment longitudinal performance during the platoon lane merging, using the RTTP alongside the car following strategy. 131
5.27 Merging vehicle Lateral performance during the platoon lane merging, using the RTTP approach alongside the car following strategy in a mixed test environment. 132
5.28 Mixed test environment platoon lane merging test scenario using three vehicles, with RTTP and the car following strategy under the IoTAC project framework. 134
5.29 Longitudinal performance of Platoon A and the merging vehicle during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy. 135
5.30 Longitudinal performance of Platoon B during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy in a mixed test environment. 136
5.31 Lateral performance of the merging vehicle during the platoon lane merging, using the RTTP approach alongside the car following strategy in a mixed test environment. 137

List of Tables

4.1 Comparative between HYTP and RTTP 66
4.2 Vehicles characteristics and MPC constrains 78
4.3 Advantages and disadvantages of car following and HYTP. 79
4.4 PMM messages description. 93
4.5 PCM messages description. 94

5.1 Car following configuration parameters for each platform. 102
5.2 Car following mean errors of each test. 108
5.3 Performance of the SerIoT system in the fleet management and smart intersection scenarios. 122
5.4 Mean error comparative of the merging vehicles during the platoon lane merging maneuver corresponding to each test. 138

LIST OF ACRONYMS

ACC	Adaptive Cruise Control
AD	Automated Driving
AUDRIC	AUtomated Driving Core
AV	Automated Vehicle
BOBYQA	Bound Optimization BY Quadratic Approximation
BSM	Basic Safety Message
C2C-CC	Car-2-Car Connected Consortium
CACC	Cooperative Adaptive Cruise Control
CAV	Connected Automated Vehicle
CAM	Cooperative Awareness Message
CCAM	Cooperative, Connected, Automated Mobility
C-ITS	Cooperative Intelligent Transportation Systems
CS	Control Station
C-V2X	Cellular Vehicular To Everything
DENM	Distributed Environmental Notification Message
DoS	Denial-of-Service
DSRC	Dedicated Short Range Communications
ECU	Electronic Control Unit
EMT	Empresa Municipal de Transporte de Madrid
ETSI	European Telecommunications Standards Institute
FEAM	Front-End Access Control
FSM	Finite State Machine
GN	Geo-Networking Messages

GNNS	Global Navigation Satellite Systems
HMI	Human-Machine Interface
HYTP	Hybrid Trajectory Planning
ILP	Integer Linear Programming
INRIA	Institut National de Recherche en Informatique et en Automatique
IoT	Internet of Things
ITS-G5	Intelligent Transportation System-G5
KSG	Kaspersky Security Gateway
LiDAR	Light Detection and Ranging
LQR	Linear Quadratic Regulation
MEC	Multi-access Edge Computing
MPC	Model Predictive Control
MQTT	Message Queuing Telemetry Transport
OBU	On-Board Units
OFDM	Orthogonal Frequency Division Multiplexing
OxTS	Oxford Technical Solutions
PCM	Platoon Control Messages
PD	Proportional Derivative
PID	Proportional-Integral-Derivative
PHY	Physical Layer
PMM	Platoon Management Messages
RADAR	Radio Detection And Ranging
RITS	Robotic For Intelligent Transportation Systems
RMS	Run Time Monitoring System
ROS	Robotic Operating System
RSU	Road Side Unit
RTTP	Real-Time Trajectory Planning
SAE	Society of Automotive Engineers
SDN	Software-Defined Networking
SLAM	Simultaneous Localization and Mapping

SMC	Sliding Mode Control
SoA	State of the Art
SPAT	Signal Phase and timing
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
V2I	Vehicle to Infrastructure
V2N	Vehicle to Network
V2P	Vehicle to Pedestrian
V2V	Vehicle to Vehicle
V2X	Vehicle to Everything
VPN	Virtual Private Network

LIST OF SYMBOLS

$G_{pfi-ACC}(s)$	Group of block comprising the ACC
$G_{pfi-CACC}(s)$	Group of block comprising the CACC
$Gp(s)$	Vehicle position transfer function
$G(s)$	Vehicle speed transfer function
V_{ref}	Car following speed reference
$H(s)$	Spacing policy
d_{std}	Standstill distance
h	Time gap
$v(t)$	Ego-vehicle speed
$F(s)$	Feed-forward filter
$D(s)$	Car following communication delay Block
$C(s)$	Car following controller block
w_{gc}	Gain-cross frequency
L_{∞}	String stability criteria
K_p	Proportional component of the PD controller
K_d	Derivative component of the PD controller
Φ	Phase merging
$d(t)$	Distance among vehicles
d_{final}	Final distance for gap opening/closure operations
v_{final}	Final speed for gap opening/closure operations
h_{final}	Final time gap for gap opening/closure operations
G^n	Geometric curvature
C^n	Curvature continuity

P_0	Bézier curve initial point
P_n	Bézier curve n point
$\vec{u_b}$	Straight path vector
$\vec{u_a}$	Straight path vector
P'_{lc}	Lane change intermediate point
w	Road width
D	Distance among Bézier points
P_r	Roundabout center point
R	Roundabout radius
a_o	Roundabout exit angle
a_i	Roundabout enter angle
P_{entr}	Roundabout entry point
P_{exit}	Roundabout exit point
$\vec{u_{p3}}$	Roundabout tangent vector in P_3
K_r	Roundabout curvature
a_{lat}	MPC lateral acceleration
d_{lat}	MPC lateral offset
v_{lat}	MPC lateral speed
Veh_w	MPC vehicle width
$Road_w$	MPC road width
J_{lon}	MPC longitudinal jerk
d_{lon}	MPC longitudinal distance
v_{lon}	MPC longitudinal speed
a_{lon}	MPC longitudinal acceleration
$d_{veh_{lon}}$	MPC distance constrain
v_{limit}	MPC speed limit constrain
$J(x(t), u(t))$	MPC cost function
v_{reflon}	MPC speed reference
d_{reflon}	MPC lateral reference
L	MPC weight function

M	MPC weight function
$Length$	Vehicle length
D_{ExtB}	Vehicles distance in Platoon B close/open gap operations
D_{ExtA}	Vehicle distance for gap operations by Platoon A
$Pos_{x_{Proy}}$	Projected X coordinate of the leader vehicle in the Platoon B lane
$Pos_{y_{Proy}}$	Projected Y coordinate of the leader vehicle in the Platoon B lane
d_{ref}	Reference distance in a platoons
Hs	Set of states of a Finite State Machine
s_0	Initial state of a Finite State Machine
δ	Sate transition function
D_F	Drivable space maximum distance
D_R	Drivable space rear-end distance
Ψ_0	Trajectory direction
Ψ_{veh}	Vehicle direction
Ψ_{lane}	Lane direction
D_{0-1}	Distance between control points 0 and 1
D_{2-3}	Distance between control points 2 and 3
B_L	Left bound
B_R	Right bound
pv	Points defining the vehicle rectangle
k_{max}	Maximum curvature allowed for the real-time trajectory generation
dR_{pv_j}	Distance between pv and B_R
dL_{pv_j}	Distance between pv and B_L

1 Introduction

"A dream is not only about never giving up, it's about sacrificing for what you want to achieve." - Erwin Smith

Cooperative, Connected, and Automated Mobility (CCAM) has been lauded as one of the most promising technologies in the world of transportation and it is expected to revolutionize the way of moving in the future. With the ability to reduce traffic congestion, improve road safety, and increase fuel efficiency, CCAM has the potential to transform cities and highways. However, despite its promises, there are still many technical and safety challenges that must be overcome before becomes a daily reality on the roads.

Automated Vehicles (AVs) face challenging situations, such as navigating dense traffic, coupled with the unpredictable behavior of other drivers, which demands precise decision-making and real-time processing of substantial amounts of data. The vehicle is required to make instantaneous decisions regarding the direction, speed, and distance of other vehicles on the road, while simultaneously detecting pedestrians and obstacles, and responding appropriately to varying road conditions. However, despite having access to this information, it is insufficient for safe and robust driving. The vehicle must possess the capability to gather more profound information about other road users, a task that can only be accomplished through communication systems.

The concept of Connected and Automated Vehicles (CAV) offers a potential solution for these challenges. CAV have the capabilities to communicate with each other and share information about their location, speed, and direction. By doing so, they can anticipate and respond to changes in traffic patterns, reducing congestion and improving traffic flow and safety.

Certain scenarios remain challenging for CAVs, despite advancements in the technologies. Complex road segments, such as roundabouts and on-ramp merging or lane changes, encompass a multitude of variables and unpredictable elements. These situations demand not only effective navigation from CAVs but also a performance that aligns with human passengers' expectations for natural and comfortable travel. Thus, there is a substantial journey ahead towards an optimal solution in these contexts.

Furthermore, considering the susceptibility of CCAM systems to cyber-security threats, it is imperative to develop robust measures to safeguard these vehicles from cyber-attacks. Such attacks may compromise vehicle control, leading to potential accidents and fatalities. Consequently, a comprehensive cyber-security strategy is essential to protect the diverse sensors and communication systems used by CCAM from hacking and other forms of cyber-threats.

Given these premises, the aim of this Ph.D. thesis is to develop algorithms for coordinating multiple CAVs in various urban scenarios, whilst also briefly exploring cyber-security approaches to further oversees the execution of these maneuvers. To fulfill this, a thorough analysis of the State of the Art (SoA) relating to the advances of various technologies associated with driving architectures and vehicle cooperation is conducted. The framework used for testing the maneuvers, which incorporates the driving architecture, the cyber-security approaches, the platforms, and the tests tracks is then presented. Finally, decision and control algorithms developed for cooperative maneuvers are outlined.

This work has been conducted in the CCAM group of the Industry and Mobility division at Tecnalia, a member of the Basque Research & Technology Alliance in the Basque Country, placed in Vizcaya, Spain. This Ph.D. thesis was partially supported by the Following European and Spanish Projects:

- ENABLE S3: (H2020 under grant agreement 692455) Stands for European Initiative to Enable Validation for Highly Automated Safe and Secure Systems. This project started on 01/05/2016 and ended on 31/05/2019.
- SERIOT: (H2020 under grant agreement 780139). Stands for Secure and Safe Internet of Things. This project started on 01/01/2018 and ended on 30/04/2021.
- Autolib: (Government of the Basque Country under grant agreement KK-2019), stands for Technology Preparation for Multi-vehicle Automation in the Industrial Sector. This project started on 01/03/2019 and ended on 01/06/2021.
- SHOW: (H2020 under grant agreement 875530). Stands for SHared automation Operating models for Worldwide adoption. This project started on 01/01/2020 and will end on 30/09/2024.
- IoTAC: (H2020 under grant agreement 952684). Stands for Security By Design IoT Development and Certificate Framework with Front-end Access Control. This project started on 01/09/2020 and ended on 31/08/2023.
- AUTOEV@L: (Government of the Basque Country under grant agreement KK-2021/00123), stands for Technology Evolution for Multi-vehicular Automation and Evaluation of Highly Automated Driving Functions. This project started on 01/3/2021 and ended on 31/05/2023.

Additionally, an academic collaboration with the Robotics for Intelligent Transportation Systems (RITS) team at the Institut National de Recherche en Informatique et en Automatique (INRIA) was established, leading to a research stay conducted from 01/13/2020, to 5/13/2020 which also includes a collaboration with the University of Berkeley.

The objectives of this doctoral Ph.D. were influenced by the goals of the European and Spanish projects, as well as the research stay commented before. The scope of these projects facilitated the exploration of distinct platforms and scenarios, while also aiding in the identification of key cooperative maneuvers and providing a framework for their testing.

1.1 Background and Motivation

Road safety continues to be a pressing global issue, with motor vehicles operated by humans significantly contributing to the high number of accidents. In 2023, an estimate by the National Highway Traffic Safety Administration revealed that 19,515 fatalities occurred due to traffic accidents during the first half of the year in the United States[1]. Concurrently, Europe witnessed an increase of 4% in road deaths in 2022 compared to the previous year, with a total of 20,678 casualties reported by the European Transport Safety Council[2]. This surge has been attributed to the easing of travel restrictions amidst the Covid-19 pandemic and a subsequent rise in vehicle ownership and usage.

A primary factor exacerbating road safety issues is traffic congestion. Research conducted by Gaitanidou et al. [1] established a link between traffic congestion and the frequency of road accidents across various European countries, including Italy, Spain, Greece, Belgium, and the Netherlands. Their findings suggest a higher probability of accidents during peak traffic hours and on highways with multiple lanes. Similarly, a study by Abdelaty et al. [2] in the urban areas of the United States found traffic congestion to be a contributing factor in 38% of all analyzed crashes and 74% of rear-end collisions.

These accidents not only endanger drivers and passengers but also negatively impact the economy through delays. In 2022, significant economic losses were reported in European cities due to traffic congestion. London and Berlin topped the list with losses amounting to £5.7 billion and 963 million euros respectively [3]. Moreover, in terms of time spent in traffic, London and Paris led with 156 and 138 hours per driver respectively.

Despite the concerning statistics, there has been a significant improvement in road safety in Europe since 2012, with a 22% reduction in the number of deaths. This positive trend can be attributed to several factors, such as the enhancements

[1] **Webpage**: National Highway Traffic Safety Administration → *https://www.nhtsa.gov/press-releases/2023-Q2-traffic-fatality-estimates*
[2] **Webpage**: European Transport Safety Council → *https://etsc.eu/euroadsafetydata/*

in road infrastructures, including improved signaling systems, reduced speed limits, and modifications to road segments.

Notably, since 2012, there has been a global trend towards replacing cross intersections with roundabouts. Studies have shown that such modifications can reduce fatal accidents by up to 65% [4], as roundabouts not only decrease conflict points but also requires reduced driving speeds. Nonetheless, these improvements alone are insufficient. If used incorrectly, there remains a high risk of accidents and increased traffic delays.

In particular, roundabouts, cross intersections, and highway incorporation present driving challenges due to their rules and geometry. These factors may pose difficulties for drivers, leading to incorrect usage of these infrastructures. Further, when maneuvers such as lane changes or overtaking are added to the mix, the challenges only escalate.

It is in these situations where a solution that elevates driving safety to a new level is needed. A solution capable of coordinate multiple vehicles to effectively drive through these road segments. This is where CCAM technologies emerges as a formidable solution, that is currently under development.

CCAM technologies endeavors to enhance transportation efficiency and safety. This is achieved by integrating CAV with advanced communication technologies and advance algorithms. The aim of CCAM is to establish a seamless, transportation system capable of optimizing traffic flow, mitigating congestion, and improving safety. The technologies has garnered significant attention from companies and research centers globally, with Europe being one of the principal focus. The subsequent section provides a brief overview of the evolution of these technologies in the European context.

1.1.1 From Automated Driving to Cooperative, Connected, and Automated Mobility in Europe

The evolution of Automated Driving (AD) technologies, from its inception to its application in contemporary commercial vehicles, is a significant aspect of Europe's automotive history. One of the notable milestones was the formation of the European consortium EUREKA (1986-1995) [5], which brought together over 70 industries and 120 research institutes. The consortium's primary aim was to foster integration and technological exchange between private and public companies and universities. Among its various initiatives, the Program for European Traffic with Highest Efficiency (PROMETHEUS) [6] stood out. PROMETHEUS aimed to enhance drivers' information about their environment, monitor road and driver conditions, and augment the vehicle's vision in comparison to the driver, along with active driver support.

PROMETHEUS and several other projects outside Europe marked a significant moment in the history of AD technologies, sparking numerous research and industry

initiatives. The European Union (EU) has sponsored several research initiatives regarding this topic. The focus of these projects has evolved over time, initially concentrating on the development of individual urban transportation concepts and energy efficiency enhancement, as exemplified by initiatives such as Citymobil [7] and Citymobil2 [8]. Gradually, the emphasis has shifted towards the creation of a comprehensive ecosystem capable of functioning in real traffic conditions, taking into account factors such as pedestrians, infrastructure, and cloud connectivity. This is seen in projects like L3Pilot [9] and HEADSTART [10]. As a result of this transition, the concept of AD has broadened from a vehicle-centric definition to a more comprehensive one that now refers to the entire mobility ecosystem, hence the term CCAM. CCAM encompasses all transport-related aspects, including communication technologies, aiming to create a seamless and intelligent transportation system that optimizes traffic flow, reduces congestion, and enhances safety. More details about CCAM can be found in Section 2.2.

Figure 1.1 provides a summary of the most significant EU projects active between 2015 to 2023, as extracted from the ERTRAC CCAM roadmap of 2022 [11]. These projects are categorized into four domains: 1) Highly Automated Urban Transport Systems, 2) Driver Assistance Systems and Partial Automation, 3) Infrastructure Connectivity and Cooperative Systems, and 4) Networking Coordination and Support.

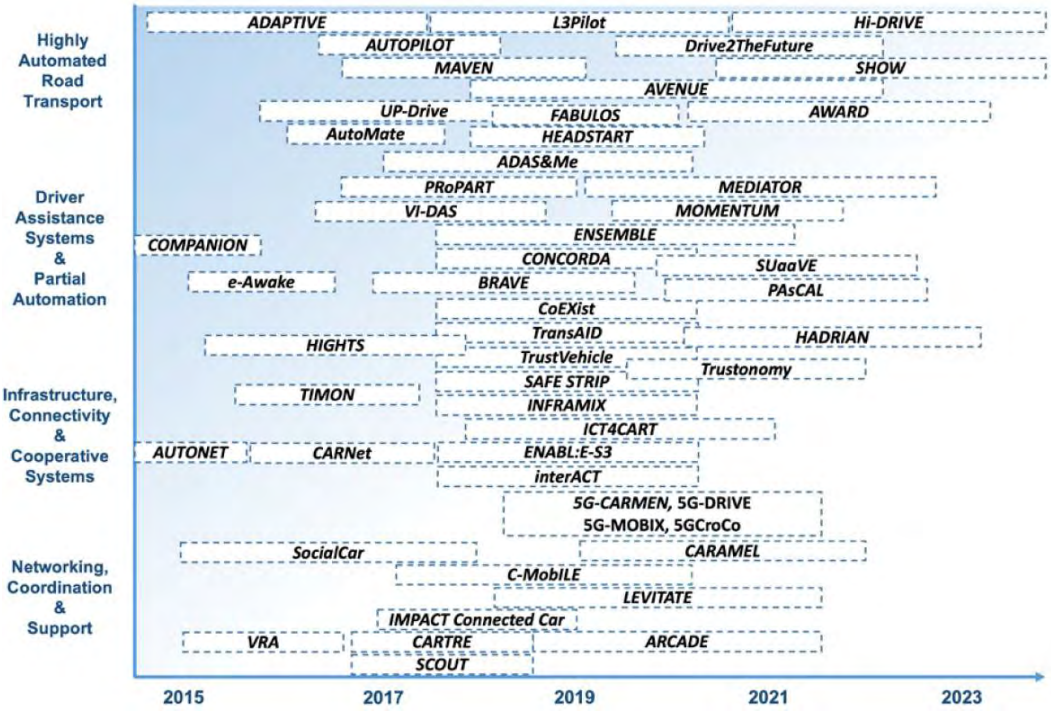


Fig. 1.1: Overview of the most relevant EU founded projects that support the CCAM technologies between 2015-2023 [11].

Significant projects such as ENSEMBLE [12] and SHOW [13] are crucial steps towards the practical realization of CCAM technologies. ENSEMBLE aims to

establish pre-standards for interoperability among trucks, platoons, and logistics solution providers, facilitating the harmonization of legal frameworks across member states. On the other hand, SHOW supports the deployment of shared, connected, and electrified automation in urban transport, thereby promoting sustainable urban mobility. Throughout the project, real-life urban demonstrations in 20 cities across Europe will integrate fleets of CAV into public transport, Demand-Responsive Transport, Mobility as a Service, and Logistics as a Service schemes. This initiative is set to be the most comprehensive pilot of CCAM in urban environments to date. The SHOW project framework facilitated part of this thesis, providing the necessary means to test platooning maneuvers to enhance operations within a bus depot.

A critical focus within the CCAM research paradigm pertains to the development of novel communication technologies, which facilitate the exchange of information among vehicles and their surrounding infrastructure. However, these advancements are not without their inherent risks, particularly concerning the susceptibility of vehicles to cyber-attacks. In response to this challenge, the EU, under the INDUSTRIAL LEADERSHIP program, initiated several projects related to the Internet of Things (IoT). Specifically, two projects, SerIoT [14] and IoTAC [15], aim to address these cyber-security concerns within the CCAM environment.

SerIoT seeks to leverage key enabling technologies to enhance the safety, reliability, and resource efficiency of next-generation IoT-connected devices. Alternatively, IoTAC aspires to develop a secure, privacy-focused IoT architecture that bolsters the resilience of IoT service environments. This thesis has been conducted within the frameworks of these two projects, providing a platform for the development and testing of CCAM maneuvers such as car following, merging, fleet management, and intersection management, all while ensuring protection against cyber-threats.

Given the pressing issues of modern road traffic and the evolution of EU initiatives, the current work draws inspiration from the need to implement CCAM technologies on roads. This research highlights the importance of executing maneuvers involving multiple CAVs in complex urban road segments. Such operations demands extensive communication among vehicles to ensure safe maneuver execution within a secure cyber framework. Exploring this domain will facilitate advancements in CCAM technologies, thereby enhancing traffic safety and reducing congestion. Moreover, this research could contribute to finding a standardized solution to a currently multifaceted problem— the exchange of information and actions required to execute maneuvers among vehicles.

1.2 Objectives

Considering the research context, the primary aim is to significantly advance the practical application of CCAM technologies, with a particular focus on the interaction among vehicles and other road users. Therefore, the principal objective of this thesis is to develop and evaluate cooperative maneuvers to facilitate the

seamless and safe navigation of multiple vehicles through various urban scenarios. This goal is further refined into the following sub-objectives:

- Design and implement decision algorithms capable of addressing different driving scenarios.
- Implement car following and merging algorithm to safely join two platoons of vehicles.
- Explore cyber-security environments to further protect the execution of the maneuvers.
- Implementation of the algorithms in different virtual and real platforms to validate the performance and adaptability of each algorithm.

1.3 Manuscript Organization

The manuscript of this Ph.D. thesis is structured as follows:

- **Chapter 2** offers a comprehensive review of cooperative maneuvers involving various agents, summarizing recent advancements in the field. This chapter examines the composition of the driving architecture, thereby providing a foundational understanding of these maneuvers. After discussing what constitutes a CAV as an individual entity, an overview of CCAM is provided, followed by an in-depth exploration of cooperative maneuvers.
- **Chapter 3** introduces the validation framework used for the development and testing of CCAM cooperative maneuver. It begins with an exposition of the driving architecture employed, which also encompasses the infrastructure along with the two cyber-security approaches investigated during the thesis, the SerIoT and IoTAC systems. Subsequently, the platforms and Test tracks for testing are presented.
- **Chapter 4** provides an examination of the car following strategy which includes an in-depth description of the control strategy used. Subsequently, the discussion shifts towards two decision methods designed for cooperative maneuvers, with an emphasis on the case of merging maneuvers. These methods are: HYbrid Trajectory Planning (HYTP) and Real-Time Trajectory Planning (RTTP). The chapter concludes with an examination of the negotiation processes required to execute maneuvers involving platoons of vehicles.
- **Chapter 5** delineates the outcomes of validation tests conducted for each maneuver elaborated in Chapter 4 and the cyber-security environments outlined in Chapter 3. The chapter starts with the presentation of simulation and real-world results for the car following maneuver using the car following strategy. This is succeeded by discussing the outcomes derived from simulation and mixed environments for roundabout and lane merging maneuvers employing the HYTP, inclusive of the maneuvers executed to test the SerIoT system. The chapter concludes by detailing the simulation and mixed environment tests

for the second decision method, the RTTP for the lane merging maneuver, incorporating the results relating to the cyber-security environment within the IoTAC system.

- **Chapter 6** summarizes the primary contributions of this Ph.D. thesis, grounded in a review of the SoA in CCAM, the development of cooperative maneuvers, and the tests conducted. Moreover, it presents recommendations and potential avenues for future research resulting from this study.

1.4 Contributions

The main contributions of this Ph.D. thesis are summarized below:

- **Contribution 1**: A cooperative maneuver framework for validating and testing cooperative maneuvers with real and virtual vehicles. In this framework, it is also incorporated a mix test environment so the maneuvers can be executed more safely.
- **Contribution 2**: An architecture that facilitate the implementation of different cyber-security environments, which involves IoT solutions to protect and oversee the correct execution of different maneuvers.
- **Contribution 3**: A combination of a car following control algorithm with two decision methods, the HYTP and the RTTP to merge two platoons into one lane.
- **Contribution 4**: A decision-making algorithm, based on a Finite State Machine (FSM) that can deal with different driving maneuvers, in both styles, cooperative and non-cooperative. With the main focus on the *lane keeping*, *car following*, and *merging* states.
- **Contribution 5**: A negotiation algorithm, involving V2X communication to execute a platoon formation, dismiss, and merge. Which uses Cooperative Awareness Messages (CAM), so as Geo-Networking messages (GN), to exchange information about the vehicle states and intentions.

1.5 Dissemination

1.5.1 Conferences

- **C1**: **Hidalgo, C.**, Lattarulo, R., Pérez, J., & Asua, E. (2019, February). Intelligent Longitudinal Merging Maneuver at Roundabouts Based on Hybrid Planning Approach. In Computer Aided Systems Theory–EUROCAST 2019: 17th International Conference, Las Palmas de Gran Canaria, Spain, February 17–22, 2019, Revised Selected Papers, Part II 17 (pp. 129-136). Springer International Publishing.
- **C2**: **Hidalgo, C.**, Lattarulo, R., Pérez, J., & Asua, E. (2019, November). Hybrid trajectory planning approach for roundabout merging scenarios. In

2019 IEEE International Conference on Connected Vehicles and expo (ICCVE) (pp. 1-6). IEEE, Graz, Austria.

- **C3**: Lattarulo, R., **Hidalgo, C.**, Arizala, A., & Perez, J. (2021, September). Audric2: A modular and highly interconnected automated driving framework focus on decision making and vehicle control. In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC) (pp. 763-769). IEEE, Indianapolis, United States.
- **C4**: Vaca-Recalde, M. E., **Hidalgo, C.**, Matute, J. A., Lattarulo, R., Murgoitio, J., Pérez, J., Vicente, C., Iturrioz, I., Camacho, A., & Hernández-Galán, C. (2022, November). Spanish approach for the Smart Digital Ports through highly automated logistics. Transportation Research Procedia, 72, 155-162, Lisbon, Portugal.
- **C5**: Szántó, M., **Hidalgo, C.**, González L., (2023, February). Low Latency Map Updates for Automated Vehicle Maneuvering. In 2023 Workshop on the Advances of Information Technology, Budapest, Hungary.

1.5.2 Journals

- **J1: Hidalgo, C.**, Marcano, M., Fernández, G., & Pérez, J. M. (2019). Maniobras cooperativas aplicadas a vehículos automatizados en en-tornos virtuales y reales. In Revista Iberoamericana de Automática e Informática industrial.
- **J2**: **Hidalgo, C.**, Lattarulo, R., Flores, C., & Rastelli, J. P. (2021). Platoon Merging Approach Based on Hybrid Trajectory Planning and CACC Strategies. Sensors, 21(8), 2626.
- **J3**: **Hidalgo, C.**, Vaca, M., Nowak, M. P., Frölich, P., Reed, M., Al-Naday, M., Mpatziakas, A., Protogerou, A., Drosou, A.,& Tzovaras, D. (2022). Detection, control and mitigation system for secure vehicular communication. Vehicular Communications, 34, 100425.
- **J4**: Szántó, M., **Hidalgo, C.**, González, L., Pérez, J., Asua, E., & Vajta, L. (2023). Trajectory Planning of Automated Vehicles Using Real-Time Map Updates. IEEE Access.
- **J5**: Vaca, M., Marcano., M, Matute, J., **Hidalgo, C.**, Martinez, B., Bilbao, S. Hernandez, F., Murgoitio, J., Pérez, J., Santaella. (2024), Connected and intelligent framework for vehicle automation in Smart-Ports. IEEE Access.

1.5.3 Book Chapter

- **B1**: Naranjo, J. E., Talavera, E., Pérez, J., & **Hidalgo, C.** (2023). Cooperative driving. In Decision-Making Techniques for Autonomous Vehicles (pp. 245-262). Elsevier.

State of the Art

2

"It is important to draw wisdom from different places. If you take it from only one place it becomes rigid and stale." - Iroh

CCAM is a revolutionary concept that has transformed the way vehicles operate. It involves the development and implementation of advanced systems and technologies that enable vehicles to function without human intervention. From basic driver-assistance features to fully automated vehicles, these technologies encompasses a wide range of capabilities. At its core, CCAM relies on cutting-edge sensors, cameras, Radio Detection And Ranging (RADAR), and other devices to perceive the surrounding environment and make informed decisions. With these technologies, vehicles can navigate roads and highways with enhanced safety and efficiency, paving the way for a future of intelligent transportation systems.

These technologies can be traced back to the early 20th century when the first attempts at creating automated vehicles were made. However, it wasn't until the late 20th century that significant advancements started to take place. One major milestone in the history of CCAM was the introduction of Adaptive Cruise Control (ACC) in the 1990s [16]. ACC allowed vehicles to automatically adjust their speed to maintain a safe distance from the vehicle ahead, greatly enhancing driving comfort and safety. Another significant development was the introduction of lane keeping assist systems, which emerged in the early 2000s. These systems employed cameras and sensors to detect lane markings and help the vehicle stay within its lane, laying the foundation for more advanced automated driving capabilities [17].

In order to gain a thorough comprehension of the current situation surrounding CCAM, it is important to examine the levels of automation. These levels have been delineated by the Society of Automotive Engineers (SAE) and subsequently accepted by the United State Department of Transportation under the Standar SAE J3016 [18].

- **Level 0**: No automation. The vehicle is fully operated by the driver.
- **Level 1**: Driver assistance. The vehicle has one automated feature, such as acceleration or steering, but the driver is still in charge.
- **Level 2**: Partial automation. The vehicle has two automated features, such as acceleration and steering, but the driver still needs to monitor the environment and intervene if needed.

- **Level 3**: Conditional automation. The vehicle can operate automatically under certain conditions, such as on highways or in good weather, but the driver must be ready to take over when the system alerts them.
- **Level 4**: High automation. The vehicle can operate automatically within set boundaries, such as geofenced areas or specific routes, and does not require any attention or assistance from the driver.
- **Level 5**: Full automation. The vehicle can operate automatically in any situation and does not need any human input or presence. No such system exists yet, and it is unclear when or if it will be achieved.

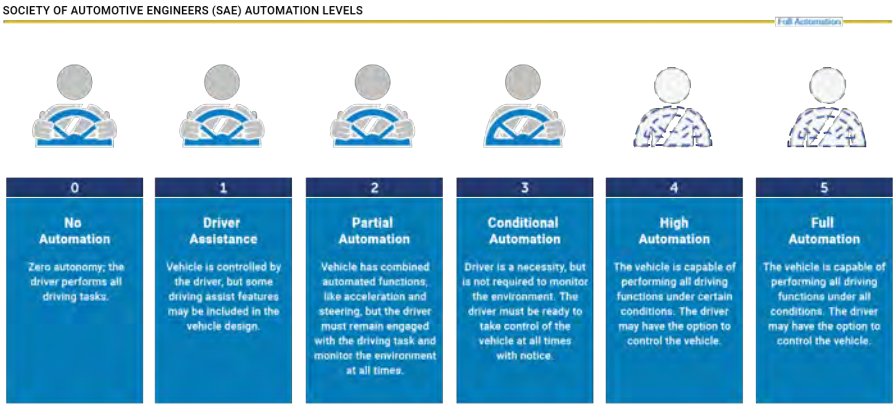


Fig. 2.1: Representation of SAE automation levels [18].

At present, the majority of CCAM technologies commercially available primarily operate at level 2. However, a discernible trend is emerging, with numerous companies initiating a shift towards levels 3 and 4, although within strict and controlled parameters. The transition between levels is nontrivial, considering each level's systems exhibit different capabilities, constraints, and assumptions. These differences significantly shape the driving architecture employed for vehicular control, forming the foundation of the vehicle's operational capacity. This capacity dictates the vehicle's ability to navigate and respond to a diverse array of potential road scenarios and conditions. Consequently, it is imperative to have a comprehensive understanding of these advances to fully understand where the technology really is.

In this way, to know in more deep these advances in the architecture, this chapter delves into the progress in various aspects that are essential to the development of CCAM technologies. First, the components that make up the vehicle's driving architecture as defined in [19] are presented: Acquisition, Perception, Communication, Decision, Control, and Actuation, with a greater emphasis on the Communication, Decision, and Control modules, as these are crucial for developing this Ph.D. Furthermore, an Infrastructure block is discussed as well. Additionally, the review addresses the various sub-modules that comprise these components, along with past definitions and implemented strategies. Once the fundamentals of individual CAV

have been established, an overview of CCAM will be provided, followed by a more detailed exploration of cooperative maneuvers.

2.1 Driving Architecture

The architecture of CCAM systems is a complex integration of various technologies designed to enable a vehicle to operate automatically. It includes hardware configurations with sensors such as RADAR, cameras, Light Detection and Ranging (LiDAR), and On-Board Units (OBU), as well as software components for localization, object recognition, path planning, and vehicle control, among others. Current developments involve similar control architectures, which were first tested on experimental vehicles at DARPA Grand Challenge [20]. Three essential tasks - perception, decision, and control - are commonly considered to enable CCAM systems [19, 21]. However, with the emergence of more complex requirements, the communication task is gaining recognition and becoming an essential aspect of the architecture [22, 23].

Figure 2.2 provides an abstract overview of a control architecture for CCAM. The main contributions of this Ph.D. thesis focus on decision, control, and communication (orange and blue blocks). Additionally, an infrastructure block is included to represent an external aspect that influences the performance of the vehicles. Although this aspect is also investigated in this thesis, it receives less emphasis compared to the other areas (yellow blocks). A description of the main systems is provided next.

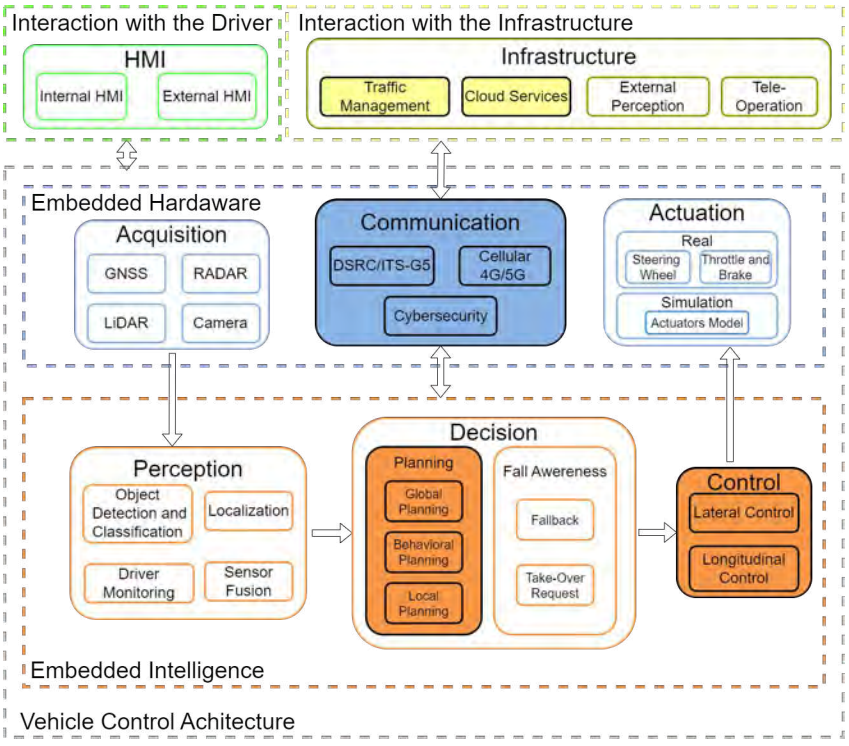


Fig. 2.2: Abstract overview of the control architecture [19].

2.1.1 Acquisition

The acquisition module collects information from in-vehicle and onboard sensors or a virtual model. It provides an initial representation of the vehicle state. In-vehicle sensors like accelerometers, gyroscopes, and others enable systems such as Anti-Lock Brake Systems and Traction Control Systems.

Onboard sensors include RADAR, LiDAR, camera, OBU, and Global Navigation Satellite Systems (GNSS). **RADAR** [24] are robust against adverse weather conditions and can estimate relative velocity. **LiDAR** [25] provides a high-resolution 3D representation of the environment, useful for object detection and localization. **Cameras** [26] collect information about the surroundings and integrate intelligent features like road obstacle detection, lane detection, and more. The **OBU** [27] uses wireless communication to obtain data and can transmit real-time messages to the other transportation systems. Through wireless interactions, drivers can access traffic conditions beyond their sensory abilities. **GNSS** [28] provides accurate location in combination with other technologies and is essential for timing synchronization.

2.1.2 Perception

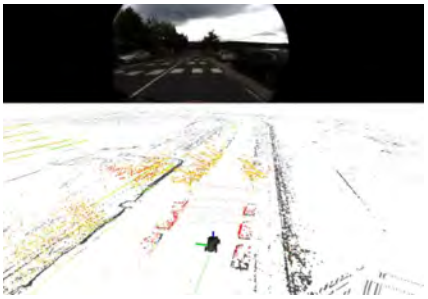
The perception module processes data from the acquisition and communication block, generating a representation of both the vehicle and its surroundings through object detection, classification, and environment recognition. Furthermore, it uses Simultaneous Localization and Mapping (SLAM) and/or GNSS systems for localization, and environment recognition, and Driver Monitoring Systems to observe the driver and/or passengers. Figure 2.3 shows an example of different perception system.

Localization is the process of determining an object's position relative to a reference map. There are two types of maps: Planar maps [29] and Point-cloud maps [30]. The selection depends on the sensor used for vehicle localization, with LiDAR and RADAR being the most common due to their superior performance at the cost of high power and processing time.

Obstacle detection and classification can be achieved using various sensors such as LiDAR, RADAR, camera, or even through communication with an OBU. LiDAR and RADAR are the most commonly used sensors in the perception system due to their excellent performance in obstacle detection. However, each sensor has its limitations [31], an therefore achieving the optimal perception of surroundings can only be accomplished through **Sensor Data Fusion** [32].

No single sensor offers the required accuracy and robustness for automated driving. However, fusing data from multiple sensors holds significant potential for designing a cost-effective localization and detection system that meets the accuracy requirements for CCAM.

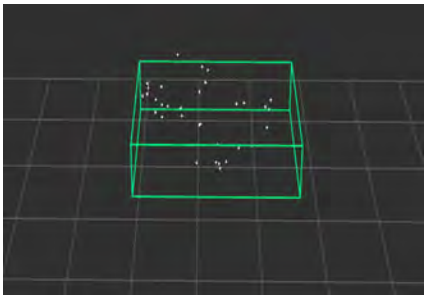
[1] **Webpage**: Medium → *https://becominghuman.ai/computer-vision-applications-in-self-driving-cars-610561e14118*



(a) SLAM and object detection with LiDAR. The real view on top and the point cloud in botton.

(b) Face recognition with camara, showing mainly the direction of the head and eyes.

(c) Object classification with camera.[1]



(d) Radar object detection shown as a green rectangle in Rviz.

Fig. 2.3: Examples of different Perception systems, RADAR, LiDAR and camera.

Relying solely on vehicle perception systems can lead to blind spots, mitigated by integrating communication systems for vehicle-road cooperation. Strategically placed road sensors provide multiple perception perspectives, offering advantages like information collection beyond line-of-sight, wider coverage, environmental resilience, and cost-effectiveness. This also enhances the accuracy and reliability of sensor-based systems [33]. This combination promise great results in the future, however is out of the scope of this thesis.

Driver monitoring systems is ca combination of various technologies to observe the driver's state and offer alerts or assistance as required. Thus, machine learning and AI methods are increasingly used to examine driver behavior patterns and produce models that precisely recognize the driver's status (the most researched in literature are drowsiness, fatigue, attention/distraction, and workload) [34].

2.1.3 Communication

The communication block is responsible for transmitting internal information and intentions of the vehicle, as well as receiving data from external sources such as other vehicles, infrastructure, pedestrians, or the cloud. This block is directly connected to the Embedded Intelligence of the vehicle, where obtains data about the ego-vehicle, obstacles or receives intentions.

Vehicular communication networks are flexible systems that ensure optimal real-time transmissions among devices in different environments, such as cities,

roads, and ports. They exchange information such as safety warnings, traffic information, and orders or movements to be carried out. These communications are typically developed as part of Cooperative Intelligent Transportation Systems (C-ITS), enabling communication among vehicles and other agents to improve the efficiency and security of control systems. By adopting a cooperative approach, vehicle communication systems can be more effective in avoiding accidents and traffic congestion compared to individual vehicle solutions [35]. This type of communication is usually known as **Vehicle to everything (V2X)** (Figure 2.4), which due to the different links that may exist can be divided into four types [36]:

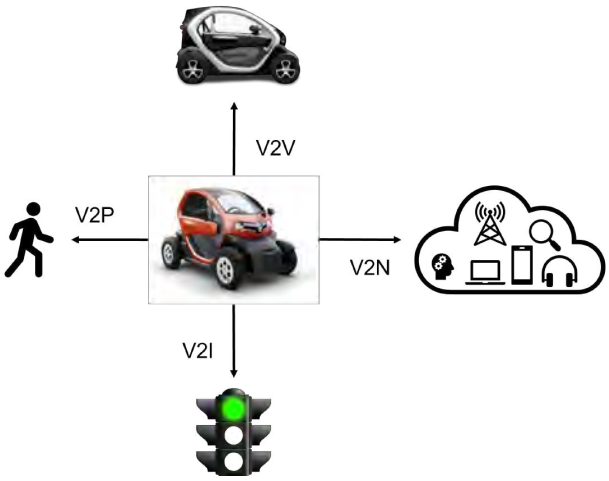


Fig. 2.4: Different communication types that composes the V2X.

- **Vehicle to Vehicle (V2V)**: This type of communication among vehicles does not require a fixed infrastructure to manage the interaction among them. It is generally used in safety applications, risk prevention, and information dissemination. Due to its mobile nature, this is a type of vehicular ad-hoc network, better known as VANETs.
- **Vehicle to Infrastructure (V2I)**: This is the type of communication among vehicles and infrastructure. It is used to disseminate information, as well as for data collection.
- **Vehicle to Pedestrian (V2P)**: This is the type of communication among vehicles and pedestrians, which in this case includes both people who walk and those who ride bicycles. It is generally used in risk prevention applications, as well as for pedestrians to obtain internet services.
- **Vehicle to Network (V2N)**: This is the type of connection among vehicles and application servers. Its main function is to provide cars with internet services.

In the subsequent discussion, focus will be directed towards an in-depth examination of the two communication technologies presently in use: Dedicated Short Range Communications (DSRC) in United States/Intelligent Transportation System-G5

(ITS-G5) in Europe, and Cellular 4G/5G, wit more emphasis in the Cellular V2X (C-V2X) technology. DSRC/ITS-G5 is considered the more mature technology, but recent studies have shown that C-V2X outperforms DSRC/ITS-G5. Furthermore, the current state of cyber-security measures within V2X communication is provided.

2.1.3.1 DSRC/C-ITS

Historically, communication standards in the United States and Europe has developed in parallel, mainly because the activities were supported by different research and development programs and promoted by different stakeholders; finally, they led to different sets of standards. These two approaches are DSCR in the United States and ITS-G5 in Europe. Still, the technical approaches of ITS-G5 and DSCR have many similarities, whereas the V2X communication systems in other regions are different.

Both protocols [37] have been defined as a lightweight, OS1 communication stack. It consists of three layers: the physical layer (PHY), the data link layer, and the facilities layer. This architecture is popular for real-time systems as it reduces protocol overhead and meets timing constraints. The system was designed to support different physical media, multi-application scenarios, and a multi-lane environment. This will guarantee a large variety of possible application areas for this technology.

DSRC is based on the WLAN standard IEEE 802.11-2012 [38], operating in the 5.9 GHz frequency band (specifically ranging from 5.825 GHz to 5.925 GHz). For the PHY layer, it uses Orthogonal Frequency Division Multiplexing with a new mode called Outside the Context of a Basic Service Set for immediate data exchange without prior control information exchange. For V2X communication, DSRC uses the Internet Protocol in combination with User Datagram Protocol (UDP) and Transmission Control Protocol (TCP). The IEEE 1609 series of standards [39] is used for direct communication among vehicles and among vehicles and Road Side Units (RSU), with the Wave Short Message Protocol [40] at the core.

It is noteworthy to emphasize that at the facilities layer, the SAE standard J2735 [41] defines the syntax and semantics of V2X messages. The Basic Safety Message (BSM) [42] is the most relevant message format, transmitting core state information about the sending vehicle. The BSM is a periodic message sent at a maximum rate of 10 Hz.

In Europe, the ITS-G5 standards, as well as DSRC use the 5.9 GHz frequency band and a similar PHY layer structure. However, the spectrum allocation is subdivided into parts A to D. ITS-G5A with 30 MHz is used for safety and traffic efficiency applications, ITS-G5B has 20 MHz for non-safety applications, and ITS-G5C is shared with the RLAN band. ITS-G5 specifies an ad hoc routing protocol called GeoNetworking [43] for multi-hop communication. It uses geographical

coordinates for addressing and forwarding, enabling packet delivery independent of the communication range of a single wireless hop. GN can also transmit IPv6 packets through an adaptation sub-layer called GN6.

At the facilities layer, V2X messages play a crucial role. The CAM [44] conveys vehicle state information periodically, similar to the BSM in DSRC. The Distributed Environmental Notification Message (DENM) [44] disseminates safety information in a geographical region. DENM transmission needs to be triggered by an application. For V2I communication, several services are defined, such as the Signal Phase and timing (SPAT) [45] message for Intersection Information Service, the MAP [45] message for Traffic Priority Service, and the In-Vehicle Information [46]. Signal control service messages are exchanged bidirectionally using Signal Request and Signal Status messages. The DENM and CAM are re-used for infrastructure-related services.

2.1.3.2 Cellular 4G/5G

Cellular 4G/5G technologies are integral in enabling V2N communication. This connectivity relies on various protocols, such as TCP or UDP. However, the standout protocol for IoT solutions is Message Queuing Telemetry Transport (MQTT). MQTT [47], a lightweight messaging protocol, was specifically developed for efficient data transfer in IoT environments. It employs a publish-subscribe pattern allowing publishers to create messages on a topic, a broker to disseminate these messages, and subscribers to receive messages pertinent to their subscribed topics. The lightweight nature of MQTT has popularized it in IoT, accommodating resource-constrained devices in communication. Within the context of CCAM applications, MQTT's functions can extend to database generation, monitoring purposes, among others.

In addition to the aforementioned applications, these technologies are also used in broader V2X applications, notably in C-V2X. C-V2X [48] is an automotive wireless communication technology that evolved from cellular communication technologies, such as 4G/5G, which has been developed and continuously improved by the Third-Generation Partnership Project. The PHY layer is based on Single Carrier Frequency Division Multiplexing Access and supports 10 or 20 MHz channels. Each channel is divided into sub-frames, resource blocks, and sub-channels [49].

With the emergence of 5G that can offer speeds up to 10 Gbps with low latency of 1 ms (for everyday cellular users), more opportunities arise in the CCAM field [50]. 5G-V2X can support more services than 4G-V2X, providing diverse capabilities, e.g. extremely high data rate, extremely low latency, extremely high reliability, high traffic density, and high capacity connections for different scenarios, such as Enhanced Mobile Broadband, Ultra-Reliable Low Latency Communications and Machine Type Communication [51]. In the CCAM field, 5G-V2X is composed by three key modules [52]: Proximity Service, Multi-access Edge Computing (MEC) and Network Slicing.

As well as DSRC and C-ITS, 5G can use the different services located in the facility layer. (eg. BSM for DSRC or CAM for ITS-G5, among others). Furthermore, it can incorporate services related to the network, expanding the possible usage of this technology in CAVs. Among few of these applications are teleoperation [53], High-definition Map Generation and Distribution [54], etc.

2.1.3.3 Cyber-security

In order to protect V2X systems, both the EU and the United State. rely on a Public Key infrastructure approach. These security management systems focus on how to manage the certificates for the C-ITS devices. A more detailed description of the United State. standard, which uses the Security Credential Management System can be found in [55], whereas the E.U standard, developed by the European Telecommunications Standards Institute (ETSI) can be found in [56].

Overall, nowadays there are numerous types of cyber-attacks. However, they can be classified based on their intention as follows:

- **Reduced confidentiality**: as a result of compromising the privacy of vehicle owners by tracking and identifying vehicles.
- **Reduced system integrity**: by making it untrustworthy, e.g., by introducing fake vehicles injecting false messages.
- **Reduced system availability**: by disrupting the objective to promote road safety, such as by creating a denial-of-service (DoS) situation.

Due to the evolving nature of V2X communication, there is concern about security issues. Being the main ones: Privacy Protection, Certificate Usage, Communication Modes, Message Handling, and System Level. A more detailed information about these issues can be found in these works [57, 58].

2.1.4 Decision

The Decision block gathers data from the perception block (localization and environment model) and the communication block (vehicle intention, infrastructure information) to generate a trajectory to be followed by the vehicle. Furthermore, oversees the correct functioning of the vehicle, executing a fallback and/or control transition strategies if a problem is detected. This module is constituted of two parts, one in charge of the **Planning** and one in charge of the **Fault Awareness**.

The **Planning module** is responsible for generating a smooth and continuous trajectory to fulfill the Dynamic Driving Task of the vehicle. It comprises three sub-modules, namely the **Global Planner**, **Behavioral Planner**, and **Local Planner**. The Global Planner determines the high-level route and path for the vehicle, while the Behavioral Planner focuses on decision-making and selecting appropriate driving behaviors. Lastly, the Local Planner ensures the generation of a detailed trajectory that considers the immediate surroundings and constraints. These sub-modules are described in sections 2.1.4.1, 2.1.4.2, and 2.1.4.3 respectively.

The **Fault Awareness module** is responsible for monitoring the proper functioning of the CAV systems, as well as the status of the driver and passengers. In the event of a system failure or driver/passenger issue, this module takes appropriate mitigation actions. There are two approaches to handling failures: the **Fallback system**, applicable to SAE levels 4 and 5, which automatically determines the appropriate action based on the failure, and the **Take-Over Request system**, applicable to SAE level 3, which decides whether control of the vehicle should be given to the driver or the automation system. The choice between these strategies depends on the level of automation being developed.

However, the discussion regarding Fault Awareness is beyond the scope of this Ph.D. thesis. Therefore, the planning module, which is one of the focal points of this thesis, providing insights into decision strategies used for cooperative maneuvers, will be further elaborated upon in the subsequent sections.

2.1.4.1 Global Planner

The Global Planner, also known as a Strategic Planner, is responsible for defining the initial trajectory. This involves determining the mission or route from the current position to the desired destination while considering factors such as safety, speed, distance, and energy efficiency. Route planning heavily relies on the environment's topology, using pre-defined maps to define the vehicle's "workspace" in terms of the goal. The obstacles considered in this stage are primarily static, such as route blockages, due to the time it takes to recalculate the route in real-time fast dynamic scenarios [59].

The route planning problem is a key task in Vehicle Route Planning, which involves optimizing the search for the shortest trajectory while considering a set of constraints [60]. In the literature, two types of algorithms are commonly discussed: Exact algorithms and Approximate algorithms [61, 62].

Exact Algorithms: The scope of these algorithms is small-scale problems. They can be divided into the following types [63]: direct Tree Search, Dynamic Programming, and Integer Linear Programming (ILP).

- **Direct Tree Search**: Is a search algorithm that explores the solution space by expanding a tree, with different possible trajectories [64]. These algorithms have strengths and versatility, but can be complex, time-consuming, and ineffective with dynamic data and a large number of dimensions.
- **Dynamic Programming**: Is an optimization method that breaks down a complex problem into smaller sub-problems [65]. While these algorithms offer versatility and efficiency in finding optimal solutions, they can encounter challenging sub-problems and consume significant memory, affecting the overall process.
- **ILP**: Is an optimization problem with a linear objective function and constraints. Unlike Linear Programming, ILP restricts some or all variables to

integer values and has a feasible region of discrete points [66]. For route planning, common ILP algorithms include set partitioning and column generation. Recent works have explored the use of Mixed ILP, which combines continuous and discrete variables to solve optimization problems [67]. ILP offers simplicity and faster computation, while Mixed ILP provides more flexibility [68].

Approximate Algorithms: Approximate algorithms can be divided into two categories, Classical Heuristic and Hybrid, where Classical Heuristic is a basic approximate algorithm that finds in a reasonable computation time a solution that is as good as possible, but not optimal, wheres Hybrid, approaches are a combination between Exact and Heuristic algorithms [60].

- **Classical Heuristic**: this method is categorized into three main categories, Constructive Heuristic, Improvement Heuristic, and Metaheuristic [69].
  - **Constructive Heuristic**: Builds routing solutions following some fixed empirical heuristic procedures. They generate a feasible solution fast and are easy to implement, however, often have a certain gap to the optimal solution.
  - **Improvement Heuristic**: Continuously improves an essential routing solution by performing a local search in the neighborhood. These algorithms are quite efficient in determining a local optimum, however, they can be easily trapped in a local optimal and the final solution's quality depends on the start point of the local search.
  - **Metaheuristic**: Provide high-level algorithm principles and are less problem-dependent. They typically bring ideas from natural phenomena or physical processes to design the optimization algorithm paradigm. They are usually efficient and have a global search ability.
- **Hybrid**: this method uses a combination of a heuristic applied to an exact algorithm. The hybrid method can be divided into Graph Search-Based and Sampling-Based.
  - **Graph Search-Based**: These approaches model the road as graphs. They search for a sequence of configuration states (position and orientation) from the initial state of the vehicle to the destination state within the feasible space of the configuration space [70]. The main graph-search algorithms used for route planning are Dijkstra and the A* family. These algorithms are efficient and can quickly find the shortest path between nodes in a graph. Additionally, they can handle continuous-variable optimization landscapes. However, if the graph is too large, it can be computationally intensive. Moreover, if it is not well-defined, it may not find the optimal path or get stuck in a local optimum.
  - **Sampling-Based**: These approaches involve randomly sampling the configuration space to solve timing constraints, typically in high-dimensional

spaces [71]. The solution obtained will converge to the optimal one as the number of samples approaches infinity. However, the computational cost can be high, especially if optimization is guaranteed, and they struggle with a dynamic environment where obstacles move over time.

2.1.4.2 Behavioral Planner

Behavioral planning, also known as maneuver planning or maneuver decision-making is in charge of identifying the possible maneuver sequences of the vehicle in order to achieve the pre-established goal. This block takes into account real-time events related to other road users, road conditions, and signals from infrastructure to produce maneuver indications for local planning.

The behavioral planner can be divided into two steps, Scenario Recognition, and Maneuver Decision, where the first one, includes the identification of the drivable Space, the motion prediction of surrounding obstacles, and the possible intention communicated by other agents. In the case of Maneuver Decision, it defines the criteria to determine which maneuvers can be performed by the vehicle, ranking them using evaluation criteria.

Scenario Recognition:

- **Drivable Space**: Is the reconstruction of a surrounding real driving environment, including the free drivable area, obstacles, and other relevant driving elements. It consists of all the static and dynamic traffic elements in the surrounding space. The three main techniques are [72]:
  - **The Grid Space**: This concept divides space into small grids and calculates each cell's occupancy probability. Uniform grid distribution simplifies sensor fusion but requires more computational and storage resources, whereas non-uniform grids reduce the number of grids and resource consumption at the cost of less smooth path planning.
  - **The Feature Space**: Represents obstacles by their coordinates and shapes, with the boundary defined by an analytic formula. It describes space continuously and geometrically using angles, edges, curves, and obstacle speed. It is useful for unpredictable indoor environments and is widely used in SLAM, serving as a mean to locate the vehicle. The feature space for CAVs incorporates traffic elements, and its construction involves tasks like road boundary and lane detection.
  - **The Topological Space**: Represents landmarks and their connections, focusing more on the connections than actual distances. Notably used in the visibility graph and Voronoi diagram, it helps find shortest paths efficiently and robustly. It may have limitations for CAVs when used online but can be highly successful when applied to a pre-defined map.

- **Motion Prediction**: Forecast the future movement of dynamic objects in a short-term time horizon, can be categorized into Physics-Based, Maneuver-Based, and Interaction-Aware motion models [73].
  - **Physics-Based motion models**: Consider that the motion of obstacles depends only on the law of physics, the most employed models the kinematic and the dynamic. Overall, these models offer reliability and simplicity but may be limited flexible, and highly dependent on the accurate parameters that define them.
  - **Maneuver-Based motion models**: Consider not only the physics but also the maneuvers that an obstacle may perform to predict the future motion of a vehicle. Maneuver-Based models offer greater flexibility than physics models and can be applied to a wider range of scenarios. However, they may require more computational resources and are highly dependent on the training data used.
  - **Interaction-Aware motion models**: These models are the most advanced, since besides the physics and the possible maneuvers, take into account the interaction among the different agents. Interaction-Aware models are the most suitable for complex driving scenarios while providing good adaptability in scenarios. However, it may be more computationally demanding than Maneuver-Based models.
- **Maneuver Negotiation**: Refers to the process by which CAVs collaborate and reach a consensus on a specific course of action in order to successfully execute a maneuver that requires mutual cooperation among them. Independent of the communication protocol or the decision-making algorithms this interaction needs to be short enough, so the traffic situation does not change too much. The negotiation can be classified into three types [74]:
  - **Independent Decisions After Information Exchange**: Is the most decentralized option in which after mutual information exchange, each vehicle's decisions are made independently. Based on the information received, models for others' movements can be improved, and sensor vision can be enlarged.
  - **Requests and Reactions**: In this approach, there are initiators and participants. Initiators send requests, which may contain their needs or suggestions. Participants assess these requests, informing the initiator of their decision, reaching and agreeing on the cooperative maneuver if the assessment is positive.
  - **Centralized**: In this type, a central unit such as an RSU, a MEC system, or a designated vehicle leader decides on maneuvers. With their higher computational power, RSUs or MEC systems can propose mandatory

maneuvers. For a leader vehicle, it needs to be pre-designated from a cluster.

Maneuver Decision: Is responsible for identifying and ranking the feasible maneuvers that the CAV may perform. These maneuvers can be classified into five types: Rule-Based, Utility-Based, Probabilistic-Based, Game Theory-Based, and Learning-Based [60].

- **Rule-Based**: Consists of statements where the system first observes the environment and then acts accordingly. The most common approaches can be divided into Logical Constraints and FSM. These method offers robustness and efficiency as they relay in low potential cost and easily adjust to the vehicle needs, however, they may present scalability issues
- **Utility-Based**: Use heuristics to evaluate different candidate maneuvers with respect to specific objectives. These approaches use cost functions to measure the level of achievement of each alternative maneuver. These approaches can be more efficient and safer, as well as more flexibility. On the other hand, these algorithms can be unpredictable due to the behavior of the cost functions, which can get stuck at a local minimum.
- **Probabilistic-Based**: In these approaches, uncertainty can occur in perception or in non-deterministic decision effects. The decision-making process is represented as a graph. Probabilistic methods can handle uncertainty and provide risk assessment to the process. However, they rely on large amounts of data for training and make assumptions that may not hold in the real world.
- **Game Theory-Based**: Involves building a decision-making tree with discrete action primitives to model vehicle behavior. The goal is to maximize the expected utility through a reward or utility function. Game theory approaches are robust and can effectively model the interaction among different agents. However, they can be computationally demanding and rely on assumptions that may not always hold true in the real world.
- **Learning-Based**: Are based on a neural network trained for a specific purpose. These approaches have a great learning capability, making them flexible and effective. However, they require powerful hardware and are sensitive to data loss.

2.1.4.3 Local Planner

This block generates a safe, smooth, and continuous trajectory with a speed profile to be tracked by the vehicle controllers. The study of this trajectory have been widely researched in the literature since the early stages of CAV development. It usually considers the dynamic and/or kinematic models of the vehicle to go from a starting position to a final one. The most relevant techniques are listed below [75, 76].

Numerical Optimization: Aim to minimize or maximize a function subject to different constrained variables. In the case of trajectory planning, this method is used to smooth previously calculated trajectories while taking into account the vehicle's kinematic. The main approaches are: Function Optimization and Model Predictive Method. These methods have high computational costs, due to the optimization process, that takes place at each motion state.

Graph Search-Based: Can be used in both Global Route Planning and Local Trajectory Planning. However, in this section, only the ones used in trajectory planning are considered, being State Lattice, Elastic Band and A-star. Graph-Based methods spend a lot of memory and rely on heavy computational costs, leading to low planning efficiency.

Geometry-based: Interpolates a previously set of waypoints to build a smooth trajectory that considers the vehicle's kinematic and dynamics, alongside the passenger's comfort. The most common methods are: Clothoids Curves and Bézier Curves. These method presents lower computational costs than both mentioned before. However, the clothoid curves, still have a significant computational cost due to the integration process, as opposed to the Bézier curves that have lower computational cost, because the curvature is defined by control points.

2.1.5 Control

This module calculates actuation commands to correct tracking errors on the trajectory defined in the decision module. The longitudinal and lateral motion control is achieved by selecting the appropriate throttle, brake, and steering wheel. Therefore, trajectory tracking involves path and speed following, which can be developed in a coupled or decoupled manner. There are different control schemes that can be applied to the trajectory tracking task. Regardless of the configuration, these strategies include feedback control without prediction, feedback control with prediction, and learning-based control [77].

- **Feedback control without prediction**: These strategies employ explicit control theories such as Proportional-Integral-Derivative (PID), Linear Quadratic Regulation (LQR), Sliding Mode Control (SMC), Fuzzy control, and H_∞ methods. These methods offer robust trajectory tracking performance under most driving conditions.
  - **PID**: The PID controller is a simple control law that considers the error variable (P for proportional), the integral (I for integral), and the derivative of the error variable (D for derivative). This controller is widely applicable but may face issues with the integral part and sensitivity to measurement noise. In certain cases, other control approaches may outperform this simple controller.
  - **Linear Quadratic Regulation**: LQR uses a linear plant model and optimal control theory to obtain the most suitable state feedback con-

troller. These algorithms can provide an control solution that minimizes the cost function, resulting in superior tracking performance. However, LQR is not suitable for systems with large control inputs leading to poor robustness.

  - **Sliding Mode Control**: SMC is a control method for nonlinear systems with parametric uncertainties and external disturbances. While SMC exhibits robustness compared to other control methods, chattering near the sliding surface remains an issue, especially in real-time applications. Additionally, SMC is derived in continuous-time, and its discrete-time behavior strongly depends on the sampling frequency.
  - **Fuzzy Control**: This heuristic approach uses semantic rules to define the system's output. It is commonly applied in systems where no mathematical model is known or obtaining models is challenging. The controller acts similarly to human behavior due to the human-like rules. However, tuning is not straightforward, and stability analysis is challenging without mathematical models. Moreover, the number of variables can make the rules unmanageable.
  - H_∞ **control**: H_∞ control is a robust approach that aims to control a plant affected by modeling uncertainties and parameter variations. It involves solving an optimization problem to minimize the H_∞-norm of a specific transfer function of the control system. The advantages of this approach are inherent robust stability and performance. However, depending on the approach, the design may result in complex high-order dynamic controllers that are challenging to implement in practice.

- **Feedback control with prediction**: The main control strategy used in this approach is Model Predictive Control (MPC). MPC offers optimal control by minimizing a performance criterion over a finite horizon, resulting in better tracking performance compared to other methods. It can effectively handle nonlinear systems and constraints by using a nonlinear model to predict future states and calculate optimal control inputs within safe limits [77]. However, there are challenges associated with accurate modeling, limitations in the prediction horizon, and computational complexity, which make real-time implementation challenging [78]. Additionally, signal delays from sensors and actuators can further impact the accuracy and control performance of MPC. Another challenge is setting the initial value for optimization, as unsuitable values can lead to failed or prolonged optimization with unpredictable calculation time.
- **Learning-Based control**: These strategies have been extensively explored because they are not dependent on a specific model and can address complex nonlinear control systems. Deep reinforcement learning has emerged as a potential solution to the limitations of modern control algorithms for trajectory

tracking control of CAVs [77]. However, these techniques have limitations, including the need for extensive training data, the potential for overfitting, and the challenge of interpreting the model's inner workings [77].

When dealing with the control problem in a decoupled manner the two main distinction are Lateral Control and Longitudinal control, each of one tackling a specific aspect.

In **Lateral Control**, the main task is to minimize the lateral displacement and angular error with respect to the reference trajectory. This control is commonly used in lane keeping and lane change applications. Lane keeping involves following the center of the lane, while lane change involves following a path that takes the vehicle from one lane to another. Different works have been conducted using the mentioned control approaches [79–81].

In terms of **Longitudinal Control**, the most common approach is to divide the controller level into an inner loop for throttle and brake control, and an outer loop for speed or acceleration tracking. Currently, the focus of research is on car following applications, where technologies such as ACC and Cooperative Adaptive Cruise Control (CACC) are used to follow a certain speed while maintaining a safe distance from surrounding vehicles. For more information, refer to Section 2.2.1. Many studies have been conducted using the aforementioned control approaches [82–84].

Throughout history, a decoupled scheme has been widely favored in the field of research. However, in recent times, a coupled scheme has gained increasing significance. This decision can be attributed to the ease of implementation associated with the decoupled scheme, where the control problem is divided into two separate parts, allowing for the consideration of one restriction at a time. On the other hand, the coupled scheme requires addressing both control restrictions simultaneously. It is important to note that while the decoupled scheme offers simplicity, dividing the problem may lead to a reduction in driving performance and comfort [85]. Noteworthy works exploring this scheme can be found in the literature, including studies [86, 87]

2.1.6 Actuation

The actuation block turns control signals into target signals for throttle, brake, and steering actuators. This can involve real actuators in a physical vehicle or a simulated vehicle model.

Real actuators, including throttle, brake, and steering wheel, can be mechanical, electronic, pneumatic, piezoelectric, thermal biomorphs, or brake-by-wire. Steering systems are either Electric Power Steering or Steer-By-Wire. Drive-By-Wire systems, incorporating electromechanical actuators and signal-based functions, are most common, improving fuel efficiency and design flexibility [88].

Simulated models represent the vehicle virtually, considering factors like wheel locations and suspension system geometry. They are cost-effective and enhance product quality. Vehicle dynamics simulation allows for integrated systems' evaluation [89]. However, numerical simulations are approximations and their accuracy depends on the model's complexity. Several methods exist for vehicle modeling [21]: Point-Mass, Geometric, Single-Track, Twin-Track, Multi-Body, Finite Element, and Hybrid. Each considers different aspects of the vehicle's behavior and is suitable for different applications, with varying levels of accuracy and computational efficiency.

2.1.7 Human-Machine Interface

The Human-Machine Interface (HMI) manages the interaction between the vehicle and its occupants. It's divided into Internal and External HMI [90].

The **Internal HMI** includes interfaces inside the vehicle. It has three components [90]: Automation HMI, that communicates the automation system's status to passengers. Vehicle HMI which uses visual, auditory, and tactile elements, displays vehicle status information and allows interaction with vehicle settings. Infotainment HMI allows not driving related activities like infotainment while ensuring safety. It uses the center console display or portable devices

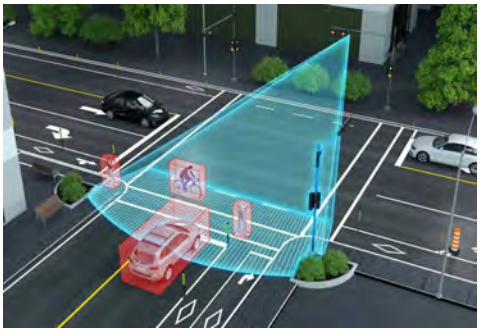
External HMI is installed on the vehicle exterior, serving as a means of communication among CAV and other VRUs [91]. Auditory HMIs assist blind or visually impaired pedestrians, whilst visual HMIs use displays, light strips, and laser projections. External HMI can enhance pedestrian safety, improve traffic flow, and facilitate right-of-way situations [92].

2.1.8 Infrastructure

The infrastructure block encompasses both physical (Figure 2.5a) and virtual (Figure 2.5b) elements outside the vehicle, ranging from road signs to large structures capable of managing heavy traffic and cloud services that complement the vehicle's driving objective. The relationship between CAV and road infrastructure has gained significant attention in recent years. However, achieving a comprehensive ecosystem among vehicles, VRUs, and infrastructure presents various challenges, including harmonization, maintenance, design, digitization, and interdisciplinary collaboration [93]. Additionally, specific applications related to infrastructure utilization are being studied concurrently, such as Traffic Management, External Perception, Cloud Services, and Tele-Operation.

Traffic Management refers to systems that are designed to manage the flow of traffic. These systems employ advanced technologies with the goal of enhancing the efficiency and safety of transportation systems by utilizing real-time data, communi-

[2] **Webpage**: Intelligent Traffic Systems → https://agtkw.com/project/intelligent-traffic-systems/
[3] **Webpage**: SAE → https://www.sae.org/news/2021/04/autonomous-vehicles-and-their-cloud-computing-networks



(a) Physical infrastructure. Composed by a traffic light and a camera.[2]

(b) Digital infrastructure, representated by a cloud where all the information is concentrated.[3]

Fig. 2.5: Infrastructure examples.

cation networks, and sophisticated algorithms[4]. The applications of these systems are diverse and encompass various areas such as: Multi-Lane-Free Flow, Intersections Management and Traffic optimization. **External Perception** involves information exchange among vehicles and infrastructure sensors to extend CAV visualization range. Various technologies facilitate this exchange, improving positioning or serving as fallback when a sensor fails [94]. **Cloud Services** [95] leverage technologies like MEC, Dynamic Resource Scheduling, and AI to enhance CAVs capabilities. The continuous data flow from various sources can improve environment representations and enrich HD maps. **Tele-Operation** allows remote operation of a CAV, providing support when a CCAM function reaches its Operational Design Domain limits. This can be implemented actively through direct control or shared control [96].

2.2 Cooperative, Connected, and Automated Mobility

The term CCAM refers to the concept of integrating connected road agents to enable automated vehicles to effectively collaborate with their environment. Various companies and organizations worldwide, particularly in Europe, are actively working towards achieving this goal. One notable organization in this field is the Car-2-Car Connected Consortium (C2C-CC), established in 2002 as part of the Amsterdam Group by vehicle manufacturers. The primary objective of the consortium is to develop European standards for C-ITS. This involves creating specifications for V2X communication operations, aligning research project ideas, and proposing work items for Standard Development Organizations. In line with the Vision Zero objective,

[4] **Webpage**:Intellistride → *https://www.intellistride.com/blog/what-is-smart-traffic-management-systems/*

the consortium has proposed the road-map depicted in Figure 2.6 [5] [6] to guide their efforts [74].

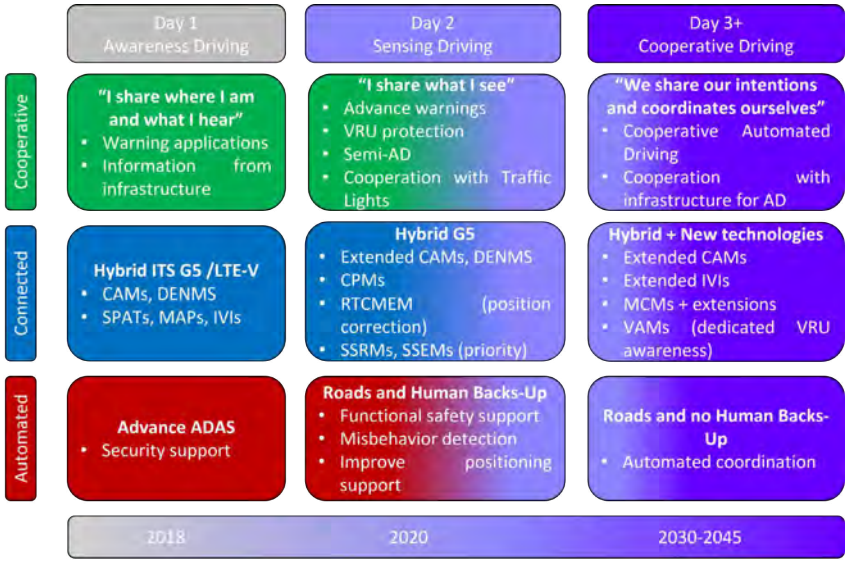


Fig. 2.6: C2C-CC Road-map [74].

- **Day 1 services**: Focus on the exchange of information to enhance foresighted driving. These services are already deployed, leveraging back-end connectivity to enable information sharing among vehicles, while Advanced Driver Assistance Systems continue to gain popularity.
- **Day 2 services**: Aim to improve service quality and enable the sharing of perception and awareness information. The consortium is currently working on version 2.0 of their Basic System Profile, which targets Day 2 services and aims to facilitate deployment.
- **Day 3+ services**: Encompass more sophisticated functionalities, such as sharing intentions, supporting negotiation and cooperation among vehicles, and paving the way towards cooperative accident-free automated driving. These services would fulfill the requirements of SAE automation levels 4 and 5; however, significant progress is still needed before these services can be fully realized. The work presented in this thesis seeks to contribute to the realization of Day 3+ services by incorporating strategies for collaboration among vehicles and infrastructures.

To achieve the desired ecosystem, it is crucial for vehicles to engage in **cooperative maneuvers**, wherein they negotiate and coordinate joint maneuvers to enhance overall utility [74]. Cooperative maneuvers involve a minimum of two vehicles, each required to perform at least one action beyond simply maintaining their current

[5] **Webpage**: C2C-CC→https://www.car-2-car.org/fileadmin/downloads/PDFs/roadmap/Roadmap-2020-figure.pdf

[6] **Webpage**: ACEA→https://www.itu.int/en/ITU-D/Regional-Presence/Europe/Documents/Events/2018/5G20Greece/Session20820Joost20Vantomme.pdf

mobility state. It is important to note that the mere exchange of information does not qualify as a maneuver, as it lacks actual driving actions. Cooperation can be achieved through two approaches: Decentralized and Centralized.

- **Decentralized**: Traffic participants are considered equal entities that broadcast information, eliminating the need for a group leader or roadside infrastructure to route information. Each vehicle independently computes its own maneuver algorithms and processes incoming requests from other vehicles. Decentralized approaches offer advantages such as lower computational costs and faster convergence speed [97]. However, they may require more sophisticated coordination mechanisms to ensure seamless collaboration among all agents [98].
- **Centralized**: Information is disseminated through a centralized infrastructure along the road or a designated group leader. In this case, only one entity computes maneuver algorithms for multiple vehicles. This approach can be advantageous in terms of having a unified control center, potentially resulting in more efficient decision-making. However, it also introduces complexity and requires further investigation to fully understand its implications [74]. Additionally, relying solely on traffic infrastructure's availability can be costly and time-consuming to establish.

Studies suggest that the optimal option for cooperative maneuvers in C-ITS is a hybrid approach, where vehicles coordinate independently while centralized entities, such as MEC servers, provide proposals to improve maneuver performance [99]. However, the coexistence of these approaches remains an open issue that requires further investigation [74].

Regarding information exchange, there is currently no clear guideline to follow. ETSI is actively working on defining a Maneuver Coordination Message that facilitates the exchange of planned trajectories and enables driving coordination[7]. However, as of the writing of this report, this message format is still under development, and no standard has been issued. Consequently, research in this area primarily focuses on defining use cases, architectures, and exchanged data [100, 101]. The main challenges revolve around bandwidth utilization, robustness against duplicated instructions, addressing various scenarios covered by the standard, and ensuring safety and comfort [100].

Additionally, ETSI has standardized a communication protocol for truck platoon operations through the EU ENSEMBLE project [12, 102]. This protocol employs a request and reaction scheme, defining two messages to facilitate the maneuver. Further details can be found in the International Organization for Standardization (ISO) 4272 document [103]. However, the implementation of such messages has not yet been adopted in commercial devices or vehicles other than trucks.

[7] **Webpage**: ETSI TR 103 439 →https://www.etsi.org/deliver/etsi-tr/103400-103499/103439/02.01.01-60/tr-103439v020101p.pdf

Regardless of the cooperation approach or the communication standard, there are numerous possible cooperative maneuvers in different situations. Some examples of these maneuvers include **Car Following**, **Intersection Management**, **Cooperative Overtaking**, and **Cooperative Merging**. These maneuvers can be carried out using different approaches, and exploring them further contributes to the understanding of cooperative systems.

2.2.1 Car Following

Car following [104] refers to the process of controlling the longitudinal behavior of a vehicle by observing or measuring the relative motion in relation to the preceding vehicle within the same driving lane. The scientific community aims to model this behavior to predict the motion of a vehicle driven by a human in a vehicular stream. This is achieved by studying the trajectory over time of the preceding vehicle (i-1) and the resulting spacing between both vehicles.

From the first studies, the Intelligent Driver Models [105, 106] are the most relevant, on which modern ACC systems are based. However, these models are not optimized for traffic flow safety or stability, but rather aim to closely imitate human behavior.

With CCAM technologies it is possible to establish an optimal reference car following model that enhances traffic performance. These reference models, commonly known as **Spacing Policies**, propose the ideal spacing between vehicles in the same lane, contributing to improved traffic flow safety and stability.

Vehicle **Spacing Policies** utilize both ego and predecessor vehicle variables, with a minority of approaches incorporating other external variables. By understanding the specific spacing policy that the members of the vehicle formation are adhering to, it becomes possible to model the entirety of the formation's behavior.

Spacing policies are designed to achieve specific performance goals. These goals include increasing traffic throughput, ensuring safety, providing comfort, and maintaining string stability. They can be classified in two types, **Constant Clearance** and **Variable Spacing**.

- **Constant Clearance**: This strategy maintains a fixed distance between vehicles regardless of their speed. It is ideal for tightly-coupled strings and significantly increases highway capacity [107]. However, it requires dedicated lanes and low-latency communication among vehicles for control stability.
- **Variable Spacing**: This strategy offer more flexibility in managing distances among vehicles, allowing control designers to target different performance metrics. The most employed technique is the Constant Time Gap, however, the Non-linear approaches have been gaining more relevance recently.
  - **Constant Time Gap**: Is based on the separation in time among vehicles known as time gap and closely mimics human car following behavior. It is

widely adopted by commercially available ACC systems [108]. Constant clearance can be seen as a special case of this technique, where the time gap is set to zero.

- **Non-Linear approaches**: They define spacing using non-linear functions of the vehicle's velocity or other variables. These approaches can be classified into two main groups: those based on human behavior for safety in car following [109] and those targeting objectives like string stability, traffic throughput, traffic flow stability, and fuel consumption economy [110].

Various spacing strategies defined by linear, quadratic, and power functions of the vehicle's speed have been compared, finding that the primary performance objectives targeted are an increase in traffic throughput, a guarantee of safety, and string stability [104]. The first objective pertains to the number of vehicles that can circulate over a single lane per unit of time. The second objective gauges the risk associated with the spacing policy in terms of collision probability, considering actuation lags or possible delays. The final objective, string stability, has been the focal point of most recent and advanced automated car following approaches. It is critical to note that the string stability of the system is determined by the selected spacing policy in conjunction with the car following control.

The concept of **String Stability** has been extensively researched throughout history. The most commonly adopted perspective is the performance-oriented approach [111], which provides adequate conditions for the development of string-stable interconnected control systems. This concept views a platoon as stable when any changes in the leading vehicles' behavior don't cause disruptions to the vehicles behind. The satisfaction of this condition not only ensures driver comfort and prevents head-to-tail collisions, but also allows for the potential extension of the string length without limitations. The variables under consideration for this study can include spacing error, control input, position, velocity, and acceleration. In this sense there are different types of stability, **String stability margin**, **Semi-strict** L_2, **Strict** L_2, and **Bounded string stability**.

- **String stability margin**: States that the entire stability of the string should be examined as a whole [112]. This aspect refers to the string stability margin of the vehicle of index (i) as the inverse of the product of all preceding vehicles' complementary sensitivity. This perspective allows for the examination of mixed strings, wherein manually driven and ACC vehicles are randomly positioned within the same string.
- **Semi-strict** L_2: Asserts that any vehicle within the controlled string should not amplify the output of the leading vehicle for the string to be semi-strictly stable [113]. This approach is primarily employed in automated car following control systems, where the ego-vehicle takes into account other vehicles besides its immediate predecessor in its control strategy.

- **Strict** L_2: Stipulates that each vehicle should not amplify the output of its immediately preceding vehicle [113].
- **Bounded string stability**: States that the spacing error of any vehicle within the string formation should not exceed a certain limit value, irrespective of the dynamics between adjacent vehicles [114].

The addition of V2V communication links to the control structure of each vehicle not only facilitates shorter inter-distances but also enables quicker reactions to state perturbations in the leading vehicle. This enhancement allows for improved string stability. These factors contribute to why CACC and platooning have attracted extensive research, showing promising results in addressing road mobility issues.

CACC technology enhances the capabilities of standard ACC by using V2X communication. This enables the exchange of supplementary data among vehicles, facilitating the synchronization of velocities. Consequently, this mitigates both the frequency and magnitude of acceleration and deceleration events, thereby preventing scenarios that could potentially escalate into critical situations [98]. Furthermore, the integration of CACC has the potential to facilitate the application known as platooning.

2.2.1.1 Platooning

Platooning [115] is an application of the C-ITS wherein a group of vehicles execute collective travel while maintaining minimal inter-vehicular distance. This arrangement can potentially augment traffic capacity, thereby facilitating enhanced traffic management and diminished transit duration. An additional advantage is the amplification of passenger comfort and safety, achieved by eliminating scenarios of drastic acceleration or deceleration and treating the platoon vehicles as a unified entity. This system also yields significant improvements in emission performance and fuel economy.

The process of platooning involves several operations [115]:

- **Joining**: It is initiated when a vehicle seeks to become a member of a platoon.
- **Merging**: Process that involves unifying two platoons heading towards the same destination.
- **Leaving**: Operation that is set in motion when a vehicle decides to exit the platoon.
- **Splitting**: Happens when multiple vehicles intend to leave the platoon and form a new one.
- **Cut-in/Cut-out maneuvers**: A cut-in maneuver is characterized by a vehicle's intent to integrate into an existing platoon, which requires the platoon to adjust and increase their inter-vehicle spacing to accommodate the newcomer. On the other hand, a cut-out maneuver occurs when a vehicle intends to exit

the platoon. This action necessitates the remaining vehicles in the platoon to decrease their spacing following the departure of the exiting vehicle.

In the journey from ACC to CACC, and now to platooning, several control techniques have been employed by industry and research entities for the task of gap regulation. These approaches include:

- **Feedforward/feedback with classical control**: This is one of the most commonly used structures. In ACC mode, it proposes to correct the spacing error with a feedback classical controller that directly commands the vehicle actuators. If V2V communication links are available, information from forward vehicle(s) is used to further improve the ego-vehicle's response towards disturbance propagation. It constitutes a two degree-of-freedom structure composed of a feedforward and feedback controller, providing flexibility and the capability to fulfill different performance requirement [116].
- **Fuzzy**: This method attempts to mimic how human drivers perform car following based on variables of interest, grounded by the theory of fuzzy logic [117].
- **LQR**: LQR-based approaches aim to use a locally optimal Linear Quadratic controller to influence the states that are feedback [118].
- **MPC**: For car following applications, it is desired to minimize the spacing and speed error with respect to the preceding vehicle. This approach also ensures global asymptotic stability and is convenient under constrained performance [119].
- **Robust control**: Strategies grounded in robust regulation have been utilized in car following control to ensure not only desired performance for the nominal model but also rejection of uncertainties in the model [120].
- **SMC**: It is premised on the interpretation of the desired equilibrium state, where the system should remain, as a sliding surface between different structures. In the case of car following, this equilibrium point is when relative speed and spacing error are zero [121].
- **Machine learning**: It is primarily used in heterogeneous platoons, where the different dynamics of the vehicles are unknown. Several algorithms attempt to predict or fit the best plant that defines the other vehicles in the platoon, ensuring string stability [122].

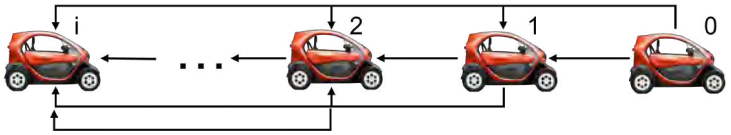
The choice of a control structure is dictated not only by the performance requirements but also by the available communication links among the vehicles within the string. The communication topology plays a crucial role, as some control structures are better suited to handle one or more variables within the control loop. Therefore, it is noteworthy that different network topologies (see Figure 2.7) are used in platooning, each optimized for specific control structures and performance requirements.

(a) Centralized control topology. The information is managed by the leader.

(b) Predecessor-Only following topology. The information used is the one obtained from the front vehicle.

(c) Bidirectional topology. The information goes in tow directions between predecesor vehicles.

(d) Leader-predecessor following topology. The information used correspond to the one of the vehicle in front and the leader.



(e) Multiple predecessor following topology. All the information of each vehicle is used.

Fig. 2.7: Examples of the platooning in function of the network topology.

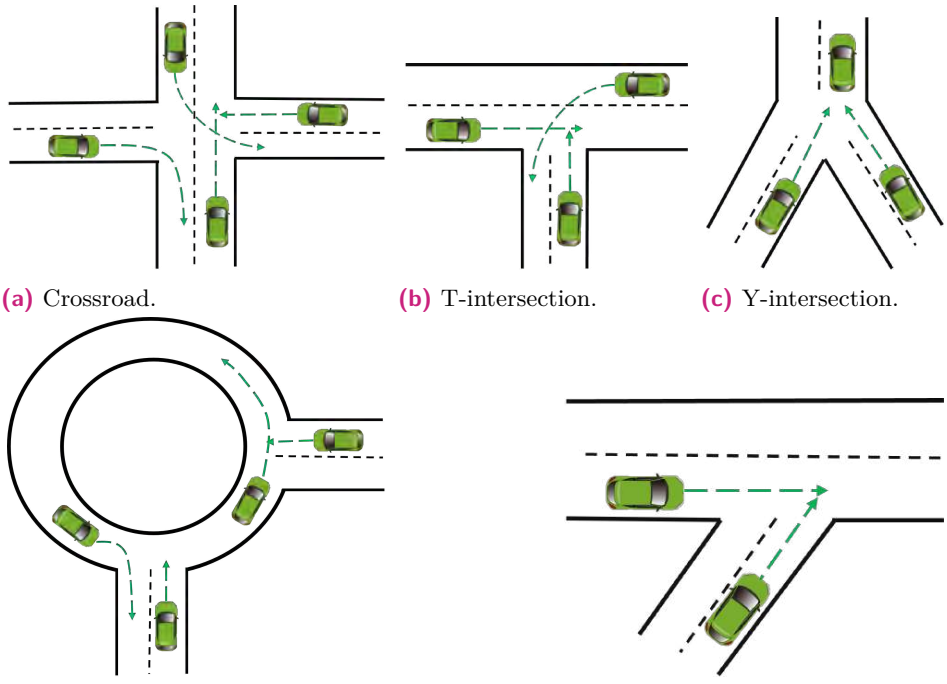
- **Centralized control**: This technique consists on having an unique control processing unit that commands each of the vehicles low level structures. It takes into consideration all variables of the string members and generates every vehicle reference states in function of the performance expected from the cooperative formation [123].
- **Predecessor-only following**: A simple approach requiring only the preceding vehicle's data and therefore the most employed. Beneficial for scalability, safety, and easy merging or splitting. However, it limits responsiveness to upstream disruptions [124].
- **Bidirectional following**: Requires two communication links for data exchange with the preceding and following vehicles. Enhances string stability by considering more vehicles, but also increases complexity due to bi-directional disruptions [125].
- **Leader-predecessor following**: Requires communication with the preceding vehicle and the leader. Allows real-time reactions to leader's changes, reducing wait time for upstream perturbations [126].

- **Multiple predecessor following**: Demands extensive communication, using data from several preceding vehicles to the leader. This achieves improved performance in case of interruptions, though its effectiveness is still debated [127].

2.2.2 Intersection Management

Efficient and safe intersection management demands the application of advanced algorithms and optimization techniques to regulate vehicular movements [128]. Due to their complex topology and rules, intersections pose significant challenges. Compared to other road segments, intersections present numerous conflict points, thereby elevating the likelihood of collisions. Notably, vehicles making left turns are at a higher risk of collision compared to those proceeding straight or making right turns.

Based on the intersection topology, they can be classified in, crossroad, Y-intersection, T-intersection, roundabout and ramp merge [129]. These intersections are illustrated in Figure 2.8 and specifically on-ramp and roundabouts are treated more in detail in Section 2.2.4 as they represent specific cases, whereas the rest of types are explained in a more generic way in this subsection.



(a) Crossroad. **(b)** T-intersection. **(c)** Y-intersection.

(d) Roundabout. **(e)** On-ramp.

Fig. 2.8: Illustration of the different types of intersections, visually demonstrating potential conflicts through the respective trajectories of each vehicle, denoted in green arrows.

Efforts to mitigate intersection-related issues have been multifaceted and can be broadly categorized into two strategies: research into traffic structures and research about CAVs [130]. The first strategy, **traffic structure research**, concentrates on the intersection's topological characteristics and signal control methods (such as

controlling traffic light timers). The implementation of diverse intersection topologies, which consequently reduces conflict points, has been demonstrated as an effective approach to enhance traffic flow and ensure secure interaction [131]. A segment of research in this category optimizes traffic light control to reinforce intersection efficiency. Some methods employ adaptive signal control algorithms to alleviate traffic congestion using real-time traffic data [132], or use learning based techniques like reinforcement learning which can process non-preset, high-dimensional sensor information and self-learn to minimize intersection delay [133]. Additional strategies include the application of Fuzzy Logic [134] to incorporate human-like decision-making into traffic light control. Despite these signal control strategies improving traffic flow to some extent, they face challenges when not all agents approaching the intersection experience equal congestion, and thus, they fail to eliminate vehicle stop delay at intersections, regardless of the traffic volume [135].

In contrast, **CAVs research** primarily focuses on vehicle algorithms for intersection crossing, encompassing individual driving strategies and cooperative driving strategies. The first type, individual driving strategies, which do not rely on V2X communication for intersection crossing, are divided into classical strategies and learning-based strategies [129].

- **Classical strategies**: Mainly focused on state machine models such as FSM and predictive methods such as MPC. Despite their widespread use due to their stability and ease of operation, these strategies are more suited to simpler scenarios rather than complex dynamic ones, because the artificially defined rules cannot adapt to all situations [136].
- **Learning-based strategies**: Has many categories, such as statistic learning, deep learning, reinforcement learning, among others. The learning-based strategies are more suited to complex dynamic scenarios but have high training costs and are challenging to semantically interpret [129].

Cooperative strategies, which use V2X communication to construct and execute a globally optimal sequence for vehicles to cross intersections, are mainly classified in centralized or decentralized.

- **Centralized**: Rely on V2I communication and require an Intersection Coordination Unit to schedule vehicles within a certain range around the intersection. The unit uses CAVs information to execute a scheduling algorithm and sends high-level decisions to the vehicles, which then individually execute low-level control. Vehicles can be treated as specific points or as groups in this case [129].
  - **Point-based methods**: Assigns varying levels of priority to individual vehicles in proximity to the intersection. This method allows vehicles with higher priority to pass through the intersection first, while lower-priority vehicles follow suit. Crucially, when the arrival times of two or more vehicles at the conflict point differ by less than a predetermined safety

threshold, the longitudinal speeds of both vehicles are adjusted to ensure safety. Numerous methodologies exist for determining vehicle priority, including rule-based method, search-based method, optimization-based method, among others.

  - **Group-based methods**: Categorize vehicles into several platoons prior to their arrival at the intersection. This strategy facilitates a safe and efficient passage through the intersection by adhering to a predetermined flow queue mode. Various approaches within this method have been examined in the literature. For instance, [137] dissected vehicles into discrete groups, framing this issue as a multi-objective cooperation problem. On the other hand, [138] organized vehicles following a standard traffic flow queue. Subsequently, an optimal control method was employed to determine the sequence of passage.

- **Decentralized**: Employs V2V communications to strategically sequence vehicles' order of intersection traversal. This method primarily operates on a first-come-first-served principle, though certain adjustments might be incorporated depending on the specific control strategy in use. Notably, much of the research in this topic has utilized Cooperative Game theory [139] and predictive control [140] as foundational methodologies.

2.2.3 Cooperative Overtaking

Cooperative Overtaking involves vehicle(s) communicating its intent to surpassing another slower-moving vehicle traveling in the same direction on a road, who can react to this request and adjust its speed or position to facilitate the safe passing of the overtaking vehicle(s) [141]. This maneuver presents significant complexity within the field of CCAM, which escalates particularly on roadways featuring an opposite lane with oncoming vehicles [142].

Before discussing the maneuver cooperatively, it is essential to outline the maneuver's non-cooperative aspects to gain insight into the current state. The overtaking maneuver comprises a suite of actions such as lane changes, vehicle acceleration, braking, and lane following. Methodologies addressing the overtaking maneuver split into two categories [143]: **Theoretical-based** and **AI-based** approaches.

- **Theoretical-based**: Involve modeling multiple scenarios and formulating the moving parts of vehicles into mathematical models. These methods often incorporate predictive control [144], non-linear optimization [145] to address the control aspects of the maneuver, and Control Algorithm [146], Game Theory [147], Fuzzy Logic [148], or FSMs [149] to handle the decision-making aspects.
- **AI-based**: Display more robustness than theoretical-based methods when handling real-life and simulation-based scenarios. AI-based methodologies

encompass Deep Learning [150], Deep Reinforcement Learning [151], Reinforcement Learning [152], and Machine Learning [153] techniques.

These approaches serve as a base to develop the maneuver, however, since they not present any type of cooperativeness, they can lead to sub-optimal and potentially hazardous situations. These scenarios may arise from conservative decision-making due to uncertainties about other participants or optimistic predictions about other participants' behavior that may be inaccurate [59]. In this context is where the **Cooperative Overtaking** scenario comes out as a viable solution since shearing their intentions reduces the uncertainties of other vehicles. In Figure 2.9 there is an example of the maneuver, where two vehicles want to surpass a slower vehicle. Through the exchange of their intentions, the green vehicle maintains its speed, while the blue vehicle in the left lane reduces its speed. This coordinated action allows the overtaking vehicles to execute the maneuver with increased safety

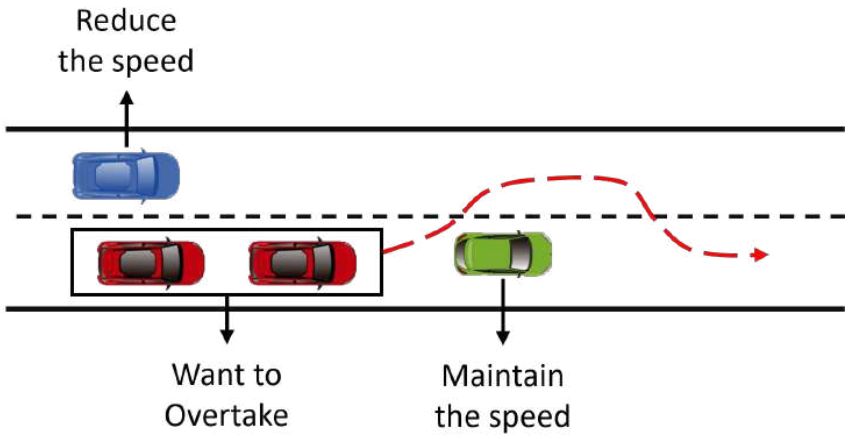


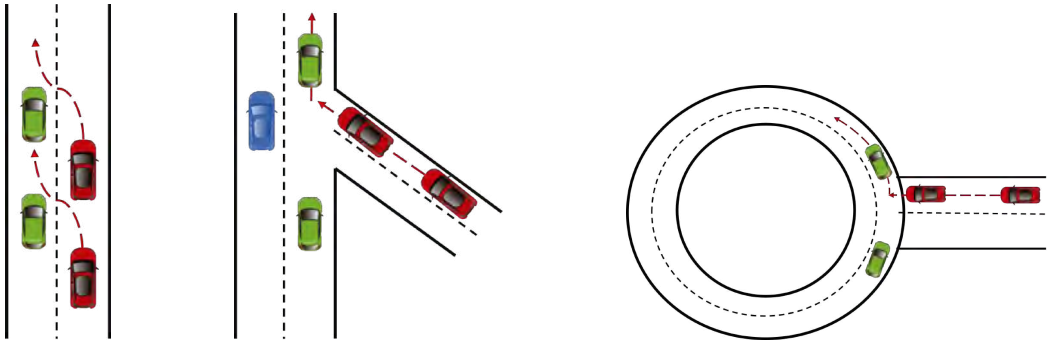
Fig. 2.9: Cooperative overtaking example, where two vehicles (illustrated in red) in the right lane intend to pass a third vehicle (illustrated in green).

In response to this problem, methods such as Cooperative Overtaking Assistance [154] and Collaborative Overtaking Assistance [155] have been developed. The first approach involves sharing the intention to overtake with other participants and executing lane change predictions. However, this system merely warns the drivers and does not take action on the overtaking vehicle. Contrarily, the second approach does take action on the overtaking vehicle, following a negotiation process with other participants to execute a safe and secure overtaking maneuver.

Lastly, an approach that is gaining interest in the research community is the cooperative overtaking maneuver for platoons. In [156] they identify the primary problem in this specific case, which lies in creating a decision-making algorithm with lane change capabilities and continuous assessments of maneuver safety while avoiding the fragmentation of the platoon into individual CAVs. They introduce a V2V-based cooperative overtaking algorithm for freeway platoons, enabling safe overtaking maneuvers without the need to disassemble and reassemble the platoon.

2.2.4 Cooperative Merging

Cooperative merging refers to a driving maneuver where vehicles cooperatively plan and execute lane merge coordination [157]. This can be seen as a multi-agent system whereby traffic participants try to reach a common goal collectively. Depending on the road segment, this maneuver can be classified in 3 types: lane merging, roundabout merging, and merging on-ramp (see Figure 2.10), being the lane merging and the roundabout merging the main scenarios of this thesis.



(a) Lane merging example.

(b) Merging on-ramp example.

(c) Roundabout merging example.

Fig. 2.10: Illustration of various types of merging scenarios, with particular emphasis on cooperative merging cases.

The **Lane Merging** scenario, as well known as lane change, involves one or multiple vehicles merging from its current lane into an adjacent lane on a roadway [158]. This requires the vehicle to assess the surrounding traffic conditions, identify a safe gap in the target lane (where the green vehicles are located in Figure 2.10a), and then execute the lane change maneuver while maintaining a safe distance from other vehicles [159]. The complexity of this scenario arises from the need to make real-time decisions based on dynamic traffic conditions, and to execute these decisions in a way that ensures the safety and comfort of all road users.

The **Roundabout Merging** scenario involves a vehicle entering a roundabout and merging into the circulating traffic [160]. This is a complex maneuver as it requires the vehicle to assess the surrounding traffic conditions, identify a safe gap in the circulating traffic, and then execute the merge maneuver while maintaining a safe distance from other vehicles [161]. This scenario is particularly challenging due to the circular nature of roundabouts and the need for vehicles to yield to circulating traffic. Improper handling of roundabout merging can lead to severe traffic congestion, travel delay, and increased fuel consumption of vehicles.

The **On-ramp Merging** scenario involves one or multiple vehicles entering a highway from a ramp and merging into the traffic flow on the main road [162]. The merging vehicle(s) must enter the major road fluidly to avoid congestion on the minor road and modify the speed of the vehicles already on the main road to

minimize the effect on that already congested main road. The mandatory nature of on-ramp merging, which requires merging within a limited distance at a safe speed, makes it one of the most challenging decision-making scenarios for CAVs [163]. Improper handling of on-ramp merging may cause a severe decrease in traffic efficiency and contribute to lower fuel economy, even increasing the collision risk.

In literature, there has been a significant evolution in the methodologies for addressing these scenarios. The progression begins from a control-oriented perspective, advances to trajectory planning, which primarily considers vehicles operating without the interference of other participants, and culminates in a more organized approach using V2X communications. In this final stage, the problem can be resolved either through communication only among vehicles or by cooperative actions. Furthermore, this stage can be implemented using either centralized or decentralized strategies.

The **control** aspect in these scenarios present distinct challenges that necessitate different methodologies for each situation. However, Fuzzy [164], Classical Control [165], and MPC [166] are the most commonly employed strategies across all scenarios. Specifically, for on-ramp merging, the goal is to prevent stop-and-go situations by smoothly integrating the vehicle into traffic, necessitating actions primarily in the longitudinal domain. In contrast, roundabouts pose more considerable challenges as vehicles must navigate circular paths, necessitating precise lateral control in addition to the longitudinal one. Similarly, lane merging requires both accurate longitudinal and lateral control as the vehicle shifts from its current lane to an adjacent one.

In terms of **trajectory planning**, several studies employ Bézier curves [167], B-spline [168] or Clothoids [169] to address these scenarios. These methods generate a smooth, continuous trajectory that safely merges the vehicle into the appropriate road segment. The on-ramp merge is less demanding, as the trajectory is more linear, comprising straight line segments. In contrast, roundabout planning is more complex, requiring the assurance of geometric and curvature continuity and smoothness in a scenario that combines straight and curved segments. Lastly, while lane merging also combines straight and curved segments like roundabouts, the curved segments in lane merging are less demanding.

The integration of V2X communications introduces an additional layer of processing to enhance maneuver performance. By leveraging data from other agents, the decision-making process gains more contextual insights, which can subsequently guide the execution of merging maneuvers. This information is then relayed to trajectory planning and, ultimately, to the control stage. The decision-making process can be executed in two modalities: connected only and cooperative.

In the scenario where only connectivity is considered, vehicles engage in information exchange, thereby enhancing their awareness of each other's locations and predicting future movements. Such approaches are inclined to optimize travel time, reduce fuel consumption, and enhance safety. However, these solutions only act on the ego vehicle, which can be counterproductive in high-density environments where

the penetration rate is low. The use of these methodologies can be found in various studies, demonstrating their applicability in addressing the challenges of vehicle merging in connected environments, being some of them, MPC [170], Game Theory [171], Virtual Platooning [172], and AI [173].

The extensive array of research conducted thus far has significantly contributed to the discovery of numerous strategies for addressing the vehicle merging problem. A common thread among these studies is the necessity to ascertain a consensus on the execution of the merging maneuver. Specifically, vehicles intending to merge must collaborate with the vehicles on the main lane to determine the sequence in which each vehicle can join the lane. This area, in particular, is currently the focal point of ongoing research efforts. The techniques for generating the merging sequence scheduling can be broadly classified into two categories, **Rule-based methods** and **Optimization-based methods** [174].

- **Rule-based methods**: Employ a pre-determined set of rules to dictate the sequence and timing of vehicles merging into traffic [175]. These methods typically involve the use of one-pair scheduling rules, such as the Shortest Processing Time and First In, First Out, to designate which vehicle has the right-of-way during a merging maneuver. This decision is based on a variety of factors, including the vehicle's speed, position, and the prevailing traffic conditions. Numerous studies have utilized time-until-arrival [176] or local-gap-optimal [177] techniques to address the scheduling problem in diverse merging scenarios.
- **Optimization-based methods**: Use mathematical optimization techniques to identify the optimal sequence and timing of vehicles merging into traffic [178]. These models may incorporate variables such as the vehicle's speed, position, and the current traffic conditions. The optimization algorithm strives to identify the optimal solution that either maximizes or minimizes a specific objective, such as minimizing the total travel time or maximizing the traffic flow [179].

The majority of existing scheduling research focuses predominantly on minimizing travel time, often neglecting the safety implications of maneuvers and the potential for multiple lanes. As identified in previous studies, the central challenges include the evaluation of strategies in continuous mixed-traffic flow, handling uncertainty in mixed traffic, and considering the presence of multiple lanes on the main road, which creates opportunities for additional lane changing maneuvers [180]. Furthermore, under mixed-traffic flow, scenarios where a combination of connected and non-connected vehicles are present can be identified. While these situations are beyond the scope of this work, which presumes all vehicles possess at least some communication capabilities, they do represent a significant challenge that warrants future research. Another complex mixed scenario arises when vehicles in platoons interact with other platoons or standalone CAVs. This scenario presents unique challenges due to

the limited flexibility of platoons caused by short-distances gaps, leading to lane blockages [181], as well as the lack of lateral actions.

2.2.4.1 Platoon Merging

In the specific context of **platoon merging**, several studies have explored different strategies. Some research, for instance, employs virtual platoons as a model for the decision-making process [182]. Others, use reinforcement learning techniques to discern the optimal method for merging two platoons [183]. Despite the promising results these algorithms produce, they heavily rely on computational resources and often overlook real-world conditions such as communication delays, negotiation, and vehicle dynamics.

Additional algorithms, like the one presented by Paranjothi, et al [184], engage in negotiations and vehicle detection to identify vehicles not intended to join the platoon, subsequently initiating splitting operations. Gao, et al [185] employed communication with a RSU to determine the safety of executing a merge, with the platoon proceeding with a split maneuver to ensure a safe merge.

In a comprehensive review of centralized and decentralized merging approaches, Li, et al [186] concluded that decentralized algorithms were more robust and demanded less in terms of communication and computation. However, they were found to be less efficient than their centralized counterparts.

Despite the significant advancements in this area, a prevalent limitation across these studies is the absence of a reliable test simulation platform. This gap reduces the applicability of these works in real-world settings. Additionally, some studies overlook key factors such as the communication link and vehicle dynamics. Moreover, the approaches to handle scheduling in merging scenarios differ significantly across studies, presenting a formidable challenge in identifying a unified solution that could contribute to standardizing this problem.

2.3 Summary and Conclusions

This chapter offers a survey of the control architecture of CCAM, shedding light on the remarkable technological advancements while also acknowledging the extant challenges. A thorough review of CCAM is also wrapped, encompassing aspects from C2C-CC consumption to the current state of cooperative maneuvers.

Various CCAM structures have been proposed over time, with the main areas of study being perception, decision, and control. Recent advancements underscore the increasing relevance of the communication module, which broadens the scope for further technological improvements. One such improvement is the infrastructure model, which has the potential to improve the driving experience.

In the realm of CCAM applications that necessitates interaction and coordination among multiple CAVs, substantial progress has been achieved. These cooperative maneuvers are crucial for optimizing traffic efficiency and safety. The interplay

between V2X communication systems, paired with sophisticated decision-making and control algorithms, is accelerating this progress. Consequently, the implementation of cooperative maneuvers is transforming the CCAM domain, revolutionizing vehicle navigation and communication.

Nonetheless, the complete implementation of CAVs and cooperative maneuvers necessitates addressing several formidable challenges. These include cybersecurity threats, concerns related to regulatory compliance, public acceptance, and infrastructural issues. In addition, the development of increasingly complex algorithms is required to effectively manage a wide range of scenarios within the cooperative scene. Furthermore, the execution of real-world tests is crucial to translate these research findings into practical applications.

This Ph.D. thesis primarily centers on these last two challenges. It aims to develop cooperative maneuver algorithms, mainly for car following and cooperative merging maneuvers that can be executed in different test scenarios, including real-world, simulated, and mixed scenarios. Accomplishing these objectives requires two key components: firstly, a modular driving architecture that can adapt to diverse scenarios, and secondly, robust decision, control, and communication algorithms, as these aspects constitute the essence of executing cooperative maneuvers.

Consequently, the following chapters will delve into the used framework as well as the developed algorithms.

3 Validation Framework

"Sometimes you gotta run before you can walk." - Tony Stark

CCAM functionalities require extensive real-world testing before they are deemed effective and stable. To alleviate the extensive hours required for real-world testing, simulations based on various scenarios often complement these tests. Consequently, reliable virtual test platforms are indispensable to reduce development time and costs.

This chapter presents the framework used for developing and evaluating algorithms for cooperative maneuvers with CAVs. The discussion begins with an overview of the AUDRIC driving architecture and how this thesis's developments are adapted within this structure.

As elucidated in Section 2.1, the driving architecture for CCAM has undergone a significant evolution, transitioning from a simple three-block structure to a more complex six-block system. This thesis further augments this architecture by integrating infrastructure interaction, a component that substantially enriches the driving experience. This chapter provides an in-depth exploration of this integration, in conjunction with an examination of the employed cyber-security frameworks.

Following the architectural description, the experimental set-up used for algorithm testing is introduced. During this thesis, two software environments are explored: 1) MATLAB/Simulink integrated with Dynacar and Visor 3D, and 2) the Robotic Operating System (ROS) used alongside the CARLA simulator. Comprehensive descriptions, usage methods, and the advantages and disadvantages of both environments are presented. Additionally, physical platforms, consisting of two Renault Twizy 80 and one Irizar 12-m i2eBus under the SHOW project, are described to provide a tangible perspective. A third testing method that combines both real and virtual platforms is also introduced. Finally, the proving grounds where the tests were executed are presented.

3.1 AUDRIC: AUtomated DRIving Core

In the process of this Ph.D, the driving architecture AUDRIC is used as the principal structure for research, being applicable in both real-world scenarios and simulation tests. As outlined in [187], the AUDRIC framework embodies different vehicle models that provides a faithful representation of the vehicle. This model is constructed specifically to scrutinize capabilities such as real-time path generation,

control, and communication, in addition to its ability to incorporate various algorithms into an Electronic Control Unit (ECU). Furthermore, presents the following characteristics:

- A modular task structure.
- Capacity to offer safe navigation under a variety of traffic conditions without limitation to a specific maneuver.
- Heightened abilities for environmental detection and modeling, and more profound integration with V2X technologies.
- Exhibits compatibility with sophisticated simulation software that delivers detailed representation of roads, inclusive of elements such as buildings, pedestrians, other vehicles, road obstructions, and traffic signals.
- It maintains adaptability with all ground vehicles that are subject to nonholonomic constraints.

The AUDRIC framework, formulated as a C/C++ library, exhibits compatibility with a broad spectrum of software platforms such as MATLAB/Simulink, ROS, and ROS2. In the context of this thesis, ROS emerged as the primary software for the utilization of this framework, nevertheless the first developments were carried out in MATLAB/Simulink. The interoperability of AUDRIC with ROS significantly streamlines the integration process with other software entities, including Autoware and MATLAB.

The framework takes inspiration from the already mentioned six-block architectural layout presented in Figure 2.2. Prior to this thesis, the AUDRIC framework was lacking in infrastructure developments. However, this gap was addressed through the SerIoT and IoTAC projects, where this module was explored, ensuring a cyber-secure environment within these systems.

A more detailed representation of this architecture can be found in Figure 3.1 where the specific technologies used are presented, highlighting the primary contributions of this research.

The **Acquisition**, **Perception**, and **HMI** modules have been developed within the CCAM group. These modules incorporate various sensors, such as GNSS or LiDAR, to facilitate localization and obstacle detection. They also feature distinct functionalities, such as switching between automated and manual modes. While these modules are used in this thesis, their development is not a part of this work.

The **Communication** module is equipped with V2X communication capabilities, using an OBU that is built upon the ITS-G5 protocol. This system facilitates various forms of data transmission, including CAM, DENM, SPAT, and MAP, as well as GN messages. Additionally, this module incorporates capabilities for exchanging data via the internet, such as MQTT and in the context of the IoTAC project incorporate two modules to detect cyber-attacks. Note-worthily, in a simulation context, it is assumed that these messages are transmitted with zero delay.

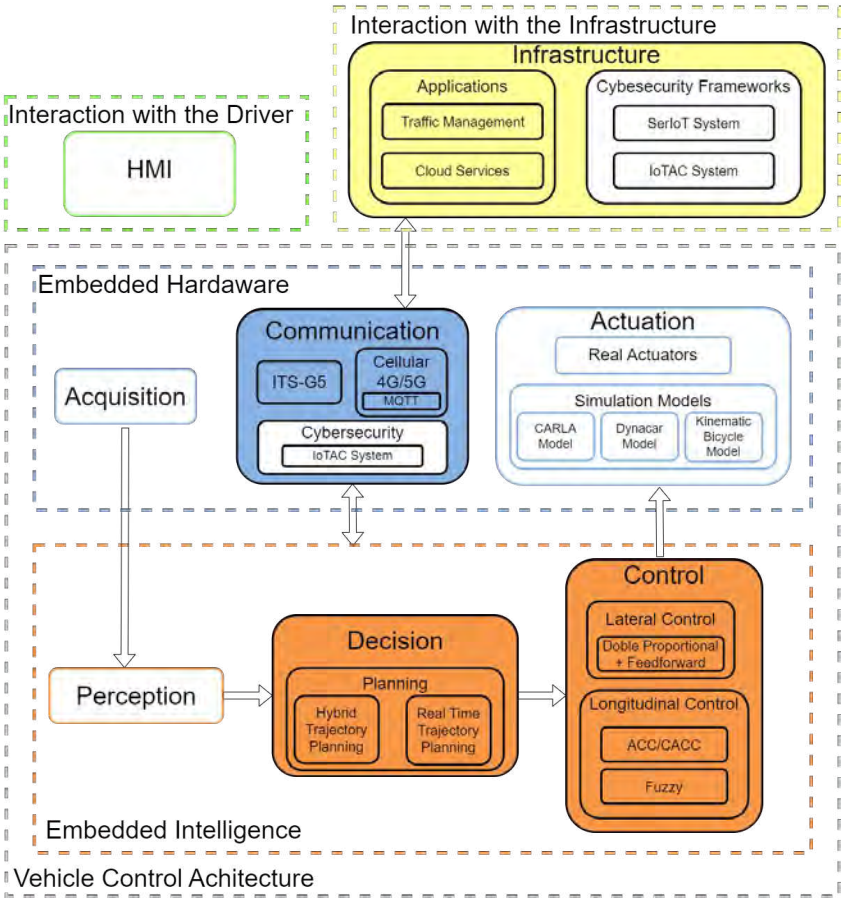


Fig. 3.1: AUDRIC driving architecture. The filled blocks represents where the main contributions of this thesis.

The **Decision** module has been designed with two distinct planning methodologies, both of which are capable of executing cooperative maneuvers. The first methodology, referred to as HYTP, and the second, known as RTTP, are grounded on the core principles of three decision sub-modules: the Global Planner, Behavioral Planner, and Local Planner. Further information regarding these methodologies can be found in Sections 4.2 and 4.3, respectively.

The **Control** module is divided into two principal components. The first component, Lateral Control, employs a double proportional plus curvature controller. This feedback controller is based on a linear combination of lateral and angular errors, supplemented by a feed-forward component which is based on curvature. The second component, Longitudinal Control, leverages a car following control algorithm (Refer to Section 4.1 for more information). Upon activation, it generates speed references, which are subsequently followed by a Fuzzy logic controller. If not activated, the speed profile of the trajectory is used. The Fuzzy controller takes into account the speed discrepancy between the reference and the actual speed of the vehicle, in addition to the vehicle's current speed and acceleration. The membership functions

are defined using triangular shapes, and the output is characterized employing single tones.

The **Actuation** module is characterized by two perspectives. The first perspective pertains to real-world applications, featuring the utilization of the Renault Twizy 80 and an Irizar 12-meter i2eBus, detailed further in Section 3.2.2. Contrarily, the second perspective is simulation-based, employing the simulated kinematic bicycle model, the dynamic model facilitated by CARLA or the Dynacar multi-body model (see Section 3.2.1).

The **Infrastructure** module, implemented within the frameworks of the SerIoT and IoTAC projects, incorporates two distinct configurations, both supported by cyber-security measures grounded on IoT solutions and adapted to the CCAM field. The first of these approaches is primarily designed to manage non-critical communication scenarios. This includes an exploration of traffic management applications, an analysis of potential vulnerabilities to cyber-attacks, and the implementation of protective measures against such attacks, all predominantly centralized within the infrastructure itself. In contrast, the second infrastructure is shaped to handle more demanding operations, such as those related to platooning. Here, the emphasis of the cyber-secure framework lies in safeguarding the infrastructure against external users, as well as investigating the potential impact of the IoTAC system on the performance of platoon maneuvers, a factor that extends to both the infrastructure and the CAV. More comprehensive descriptions of these environments is presented in the subsequent sections.

3.1.1 Infrastructure with SerIoT framework

The infrastructure presented incorporates traffic management services, primarily focused on intersection management and fleet management. This infrastructure employs RSUs, which are distributed across roads, to gather vehicle data via V2X communication. This data is then transmitted to a Control Station (CS) responsible for overseeing traffic flow and ensuring proper operation of traffic lights on roads. The CS also has the capability to dispatch recommendations or actions to vehicles in order to enhance traffic flow. Communication among the RSUs and the CS is facilitated through the TCP protocol over the internet, a link identified as the most vulnerable to cyber-attacks. To address this issue, the SerIoT system is implemented.

The SerIoT system aims to serve as a robust and reliable framework for real-time monitoring of data traffic exchanged through IoT platforms within the IoT network. It is designed to identify suspicious patterns, evaluate them, and ultimately decide on detection while simultaneously offering mitigation actions.

Building upon the AUDRIC architecture depicted in Figure 3.1, an adapted version is presented in Figure 3.2 [188]. This revised architecture is more closely tied to the infrastructure and its interconnection with the SerIoT system. This system is

developed by project partners, and therefore, one of the main contributions of this work is its integration into the infrastructure. An additional significant contribution is the development of maneuvers for testing and validation purposes. Consequently, the communication and decision blocks from AUDRIC are predominantly highlighted, as these areas were the main focus of the work conducted.

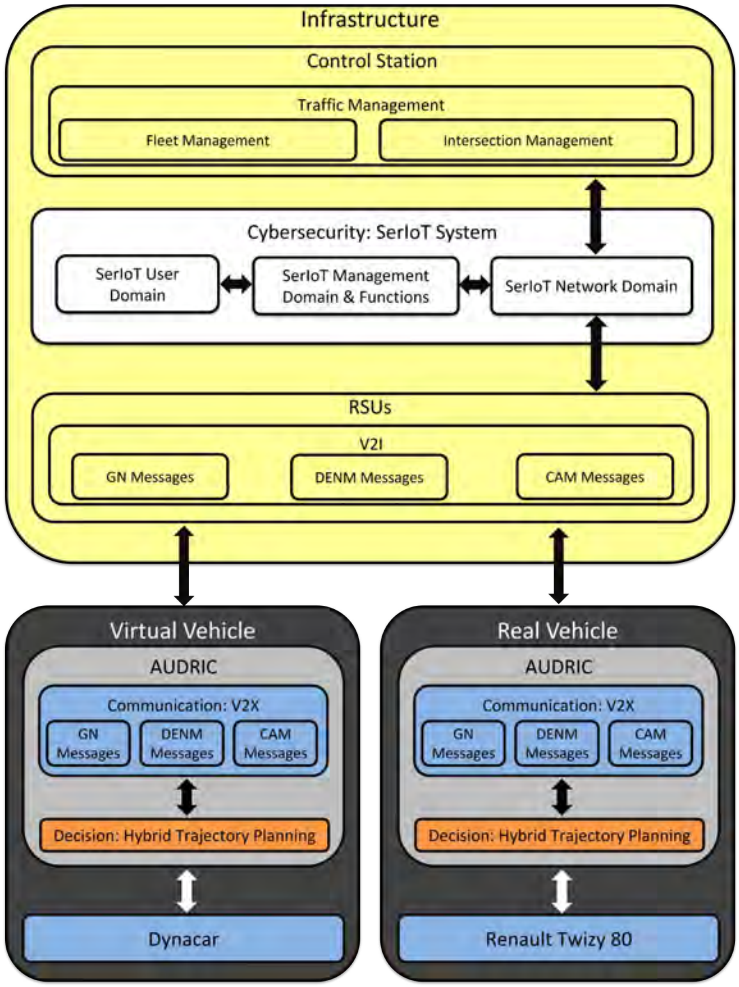


Fig. 3.2: Infrastructure scheme with the SerIoT system [188].

SerIoT system is located between the RSUs and the CS. To enable connectivity that emulates a network provider configuration, the RSUs were connected to carrier class switches that were controlled through Software-Defined Networking (SDN). The switches were interconnected through 10 Gb/s optical links as would be expected in a metropolitan area network. However, because the different SerIoT modules were hosted at different partner sites, inter-connectivity was back-hauled to the SDN switches and SerIoT components through a Virtual Private Network (VPN) connection. The VPN connection was closely monitored and found to provide consistent delay (30 ms , 0.35 ms) with no packet loss at the bit-rates used in the tests.

The SerIoT modules integrated within the project's general architecture are:

- **Network domain**: Based on SDN technology, is equipped with path optimization mechanisms. These mechanisms can be applied concurrently according to various criteria. Notably, the network domain incorporates SerIoT Fog components and serve as a mean to connect the CS with the RSUs and the rest of the SerIoT system.
- **Management domain and functions**: Include anomaly detection modules, which monitor network traffic to ensure the security of network elements and clients.
- **User domain**: Also known as network edge elements, encompasses honeypots and autopolicy modules.

For a more comprehensive understanding of the component and the SerIoT complete architecture, refer to the relevant literature [14, 188, 189].

3.1.2 Infrastructure with IoTAC framework

The second infrastructure is primarily designed to supervise platoon maneuvers, including platoon driving, engagement, merging, among others, with the objective of improving traffic flow. However, its functionality is not confined to these specific cases as it covers overall traffic management. In this model, the information exchange occurs through the MQTT protocol, where vehicles are persistently transmitting their kinematic data, sensor states, and information related to platooning. This information enables the CS, situated in the cloud, to monitor and control the CAVs, issuing instructions such as engagement, merging, or disengagement, should a vehicle be compromised. This infrastructure also allows for the presence of an operator to oversee and manage operations.

Due to the aforementioned CS operations, various concerns surges:

- Availability of the CS platform and it potential expose to attacks.
- The security of the information being exchange with the vehicles.
- The possibility of attacks in the vehicle network itself, either by having a physical entry point or by exposure to the internet connectivity.
- The correctness of the data reported by the vehicle, and the appropriate functioning of the platform.

In addressing these concerns, the IoTAC system emerges as a viable alternative to previously proposed solutions. Like the SerIoT system, the primary objective of the IoTAC system is to introduce IoT solutions to various environments, including the CCAM environment. Unlike SerIoT, however, the IoTAC system intends to extend these solutions to CAV as well. The IoTAC system is designed to define and implement a secure IoT framework that relies on Front-End Access Control (FEAM), using innovative components such as a secure element with a user card-let and a cloud-based card farm.

The AUDRIC architecture was further enhanced through the incorporation of the IoTAC system, as depicted in Figure 3.3. This adjustment chiefly concerns the arrangement of IoTAC modules within the infrastructure and the integration of embedded modules within the vehicle itself. To facilitate the integration of two distinct IoTAC modules within the real CAV, two separate devices were employed: one for the attack detection module and another for the Kaspersky Security Gateway (KSG) module. Mirroring the approach adopted in the SerIoT project, the contributions herein primarily pertain to the integration of these modules within the existing environment and the development of maneuvers for subsequent testing and validation purposes.

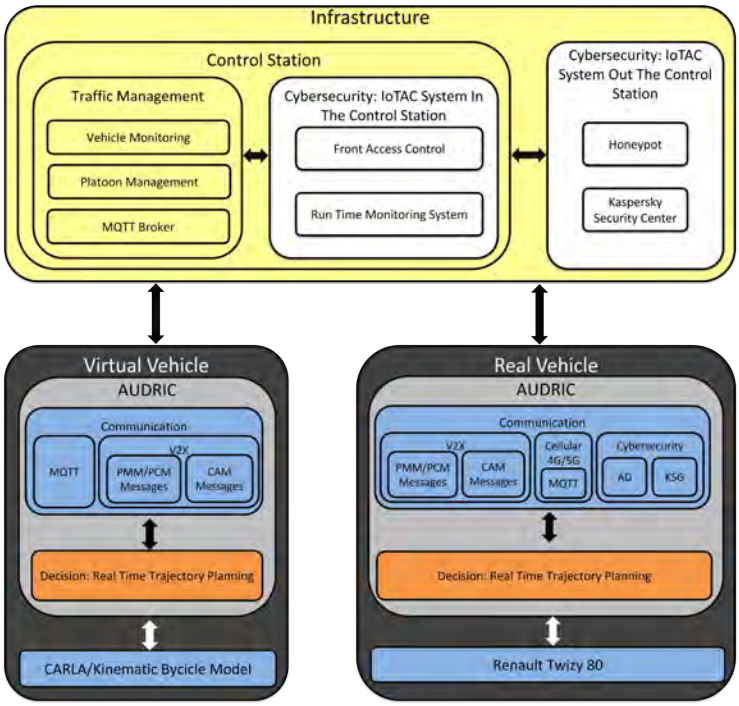


Fig. 3.3: Infrastructure scheme with the IoTAC system.

The IoTAC modules integrated in the CS are:

- **Run Time Monitoring System (RMS)**: Was installed to monitor information exchanged between the vehicle platforms and the control station, with the aim of detecting potential threats and anomalous behavior.
- **FEAM**: Was added to provide an additional level of security through user validation. This ensures that the control station, which has the ability to alter road parameters and control the formation of vehicle platooning, is only accessible to authorized users.
- **Kaspersky Security Center**: Was included in the system to provide remote monitoring of the KSG status.

The IoTAC modules integrated in the real CAV are:

- **KSG**: Was installed to function as an endpoint for attack detection, and to provide monitoring capabilities within the vehicle network.
- **Attack Detection**: Was incorporated to monitor all internal traffic of the vehicle and cover all potential entry points in the vehicle's network, detecting abnormal behavior and notifying the driver to regain control when needed.

From more information about the previously described IoTAC modules, please refer to the following articles [15, 190, 191].

3.2 Experimental set-up

For the effective testing of CCAM functionalities, a robust test environment is crucial. This environment encompasses both simulation and real-world settings, along with an integrated approach that ensures seamless interaction between the two. Additionally, it is imperative to have a well-established proving ground to ensure accurate reproduction of test conditions. This section delineates the simulation and real platforms employed for testing purposes, the methodology for their combined use, and a description of the utilized proving ground.

3.2.1 Virtual Platforms

In the process of testing CCAM functionalities, two distinct simulation environments were used. The first simulation environment used a combination of MATLAB/Simulink together with the Dynacar simulator. The second simulation environment made use of ROS in conjunction with the CARLA simulator.

3.2.1.1 MATLAB/Simulink with Dynacar and Visor 3D

MATLAB [192], an acronym for Matrix Laboratory, is a programming language and interactive platform, primarily engineered for numerical computation, visualization, and programming. The core of MATLAB is its proprietary language, which is matrix-based. Simulink, a significant component of MATLAB, serves as a graphical programming environment specifically designed for the modeling, simulation, and analysis of multi-domain dynamic systems. Furthermore, it is seamlessly integrated with MATLAB, which allows the incorporation of MATLAB algorithms and C/C++ code via S-function blocks.

As depicted in Figure 3.4, AUDRIC in Simulink consists of six main blocks, along with a HMI. This HMI includes a Panel section that enables route changes in the vehicle and functions as an automated mode on/off switch. Each of these blocks houses subsystems that contain the group's developments.

In considering the implementation of various optimization algorithms, the ACADO Toolkit [193] emerges as a suitable choice. This software, which integrates seamlessly with MATLAB, provides a comprehensive suite of algorithms designed for automatic control and dynamic optimization, rendering it an effective solution for optimal control tasks. The toolkit's versatility accommodates multiple direct optimal con-

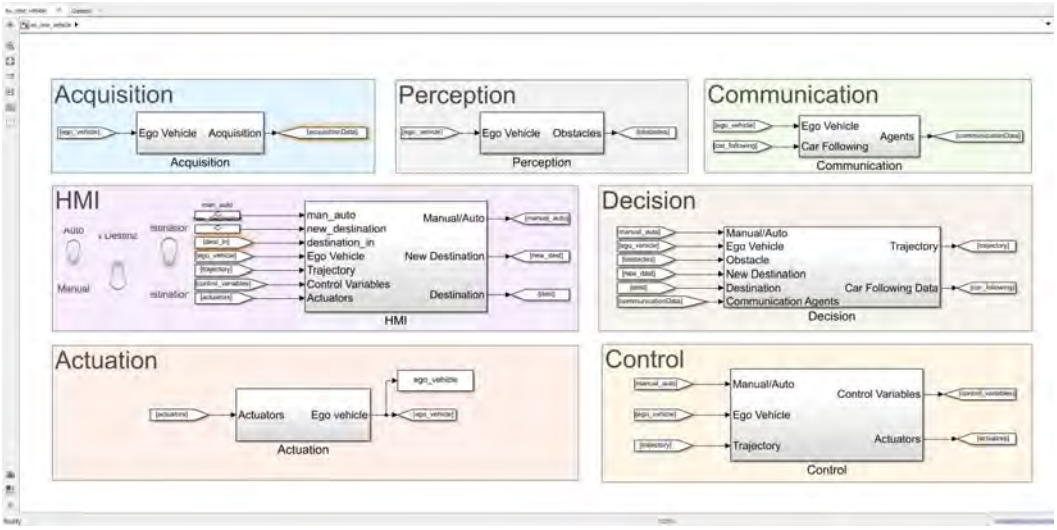


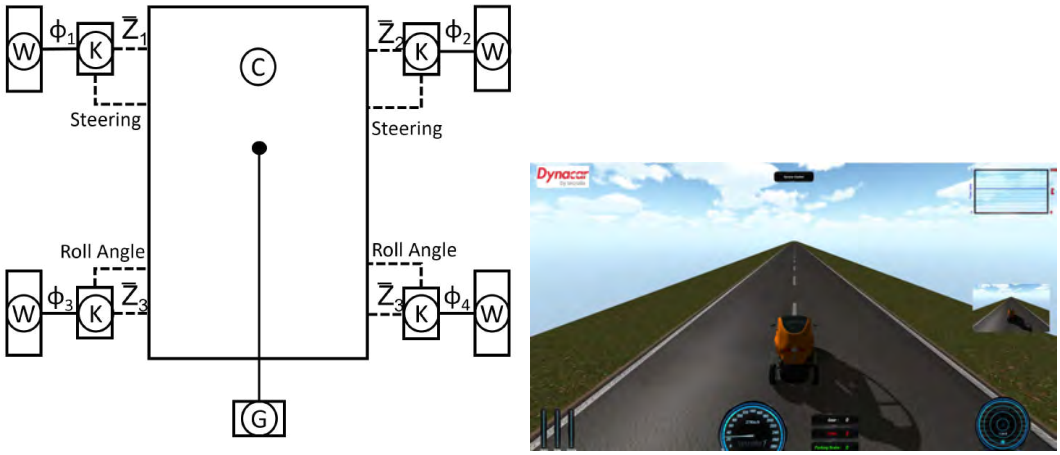
Fig. 3.4: AUDRIC architecture represented in Simulink.

trol algorithms, including MPC. Noteworthy, while the ACADO Toolkit operates as standalone C++ code, it is equipped with a user-friendly MATLAB interface, facilitating ease of use. The toolkit is primarily utilized for executing the MPC in the HYTP discussed in Section 4.2.

The Dynacar simulator [194, 195], another external software, is seamlessly integrated due to the compatibility with C/C++ code. Serving as an integrated solution, Dynacar is instrumental in the development process of electric and hybrid vehicles. It provides a comprehensive physical model of the vehicle, which is grounded in a multi-body formulation, relative coordinates, and semi-recursive equations of motion that are based on a velocity transformation. The model conceptualizes the vehicle's suspensions as macro-joints, and their behavior is characterized by utilizing lookup tables.

As illustrated in Figure 3.5a, the local Cartesian coordinates of the chassis frame are strategically located at the center of the front track width (C), whereas the cardan angles, which determine the wheel orientation with respect to the chassis frame, are situated at the steering knuckles (K). Moreover, the kinematic expressions for the macro-joints accommodate values pertaining to the position, velocity, and acceleration of the wheels (W). This model equips users with the ability to develop or integrate their control algorithms in the Matlab/Simulink environment. Notably, the Dynacar software also includes a visualization and road editor tool, termed as Visor 3D, for driving simulations as depicted in Figure 3.5.

During the initial stages of this thesis, MATLAB/Simulink software was primarily used for simulations, largely due to its prevalent use within the CCAM group and its various advantages for automated vehicle simulation. This combination offers potent tools for data analysis and synthesis, including data access, visualization, and labeling. Coupled with the Dynacar simulator, it posses a reliable vehicle model that combined with the own MATLAB/Simulink scenario simulation, algorithm



(a) Dynacar Multi-Body model.

(b) Visor 3D.

Fig. 3.5: Dynacar simulator.

design, deployment, integration and testing, provides real-world insight and reduce the need for extensive vehicle testing.

However, it is important to acknowledge a few potential drawbacks. The complexity of these tools can lead to a steep learning curve, and the cost of MATLAB/Simulink might present a financial barrier for those seeking cost-effective solutions. There might also be limitations in their simulation capabilities, especially in scenarios with multiple vehicles, which demands considerable computational power and potentially impacts simulation performance. The integration of embedded hardware can be complex, as not all sensors are compatible with the MATLAB/Simulink environment. Furthermore, the Dynacar simulator lacks customization options, making it challenging to simulate other vehicles, obstacles, and pedestrians.

In light of the need to execute more demanding simulations and hardware integration, an alternative was explored during the mid-stage of this thesis's development: the ROS software alongside the CARLA simulator.

3.2.1.2 ROS with Carla

ROS is an open-source software framework that provides a rich collection of libraries and tools, significantly assisting developers in the creation of robot applications. The framework encompasses a broad range of features, including but not limited to, hardware abstraction, device drivers, libraries, visualizers, message-passing, and package management. While ROS is predominantly used within the research community, particularly in the realm of service robotics applications, its versatility and robustness make it equally applicable to a multitude of other areas such as industrial robotics and mobile robots, including CAVs.

ROS exhibits compatibility with a broad spectrum of programming languages, inclusive of Python, C++, Lisp, and JAVA. This wide-ranging compatibility, coupled with its inherent scalability, facilitates effortless integration with other software entities, notably, the CARLA simulator.

CARLA [196], an acronym for Car Learning to Act, is an open-source simulator specifically designed to support the research, development, training, and validation of automated urban driving systems. Developed from its inception to promote advancements in automated vehicles, CARLA provides not only open-source code and protocols but also accessible digital assets. These assets include elements such as urban layouts, buildings, and vehicles, all of which are freely available for use. The simulator is equipped with a versatile specification system for sensor suites and environmental conditions. Moreover, it boasts a robust Application Programming Interface that grants users extensive control over diverse aspects of the simulation, including but not limited to, traffic generation, pedestrian behaviors, weather conditions, and sensor operations. Figure 3.6 provides a visual depiction of a typical simulation in CARLA.

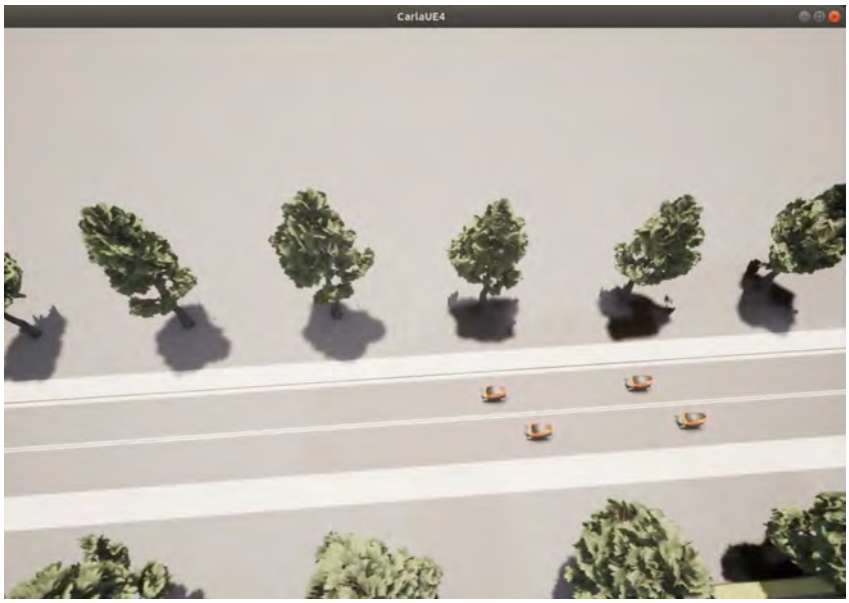
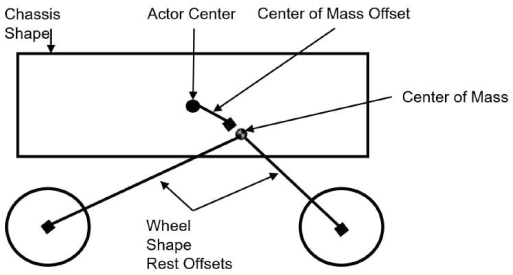


Fig. 3.6: Simulation in CARLA.

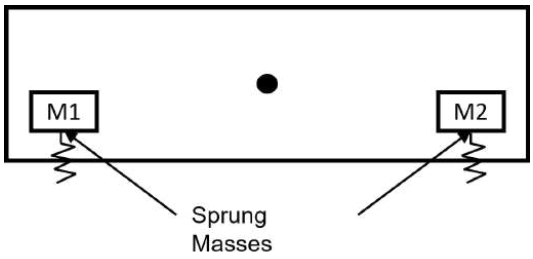
In relation to vehicle modeling, CARLA employs the Unreal plugin, which essentially serves as a wrapper for the NVIDIA PhysX vehicle [1]. This approach entails a dynamic vehicle model that conceptualizes vehicles as assemblies of sprung masses, with each sprung mass representing a suspension line equipped with corresponding wheel and tire data. Concurrently, these assemblies of sprung masses are represented as a rigid body actor, the mass, center of mass, and moment of inertia of which precisely align with the masses and coordinates of the sprung masses. This dual representation is depicted in Figure 3.7.

Another software that can be incorporated into ROS is Autoware [197], a comprehensive, open-source software platform specifically developed for CAVs and

[1] **Webpage**: NVIDIA PhysX SDK → https://docs.nvidia.com/gameworks/content/gameworkslibrary/physx/guide/Manual/Vehicles.html
[2] **Webpage**: NVIDIA PhysX SDK → *https://docs.nvidia.com/gameworks/content/ gameworkslibrary/physx/guide/Manual/Vehicles.html.*



(a) Vehicle representation as a rigid body actor with shapes for the chassis and wheels.



(b) Vehicle representation as a collection of sprung masses.

Fig. 3.7: NVIDIA PhysX vehicle model [2]

embedded systems. It offers a complete suite of modules required for automated driving, encompassing localization, detection, prediction, planning, and control. Built on the foundation of the ROS, Autoware is designed to facilitate the commercial implementation of automated driving across a wide spectrum of vehicles and applications.

This simulation environment provides numerous advantages for testing CCAM functionalities. It offers sophisticated testing and simulation tools, allowing developers to validate their software in a virtual environment before its deployment to the physical vehicles. ROS already possesses an extensive code-base for CCAM, thereby simplifying the integration and manipulation of hardware such as LiDAR, RADAR, etc., which provides an advantage over MATLAB/Simulink. Moreover, the incorporation of CARLA facilitates more comprehensive testing as it allows the simulation of not only multiple vehicles but also pedestrians and objects which is the main drawback of the Dyncar simulator.

However, this environment also possesses potential drawbacks. Its complexity can pose challenges for novices. Additionally, the current version of ROS lacks a security mechanism to prevent unauthorized access to the ROS network and interference with node communication. The absence of visualization tools further complicates the debugging process and performance evaluation of the CCAM, at difference of the extensive MATLAB/Simulink toolkit regarding this topic. Furthermore, when using a more precise vehicle model, CARLA may impose restrictions that can create obstacles in certain situations.

In summary, despite these limitations, the simulation environment offers more advantages than MATLAB/Simulink, allowing easier hardware integration and more comprehensive simulation tests. These characteristics enhance the development of this thesis, positioning it as the preferred option during the second development stage of the thesis.

3.2.2 Real Platforms

This Ph.D. thesis employs three distinct test vehicles for the proposed developments: two Renault Twizy 80 and one Irizar 12-m i2eBus. It's important to note that a forth platform, the Gulliver Shuttle, was used as a leading vehicle for platoon operations but was not subjected to algorithm testing. Figure 3.8 depicts one Twizy, the i2eBus and the Gulliver platforms.

Fig. 3.8: Real platforms used. From left to right: A 12-m Irizar i2eBus, an EMT Gulliver and a Renault Twizy 80 Under the EU project SHOW.

3.2.2.1 Renault Twizy 80

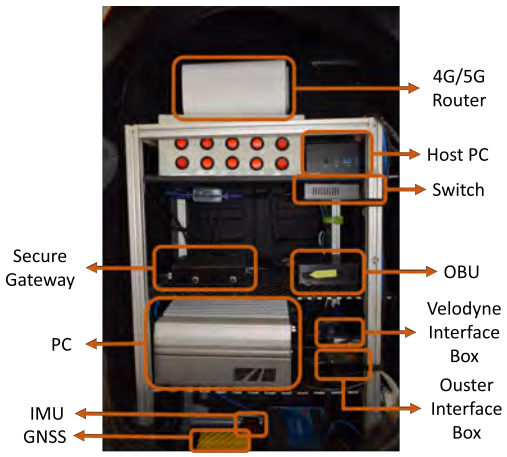
Tecnalia is home to two automated Renault Twizy 80 vehicles, which have been equipped to support automated control of the steering wheel, throttle, and brake actions. Each vehicle is capable of reaching speeds up to 80 km/h and their compact size allows for both indoor and outdoor applications. Over the course of this thesis, a variety of sensors were employed in the Twizys, that where analogue equipped with the most comprehensive array depicted in Figure 3.9.

The on-board computer, a Karbon K 700 running Linux 18.04, is the operational hub of the vehicle, hosting both ROS and Autoware with the driving architecture. This computer interfaces with a PLC via a CAN interface. In the initial stages of this research, the on-board computer operated on Windows 10 with Matlab/Simulink.

The PLC is responsible for the low-level control of Maxon motors, which are situated on the steering wheel and brake system. The motor on the steering wheel is

located centrally on the tie rod, while the motor for the braking system is attached to the brake pedal via a wire, enabling it to actuate the brake pedal based on the supplied action. Additionally, a third PLC controls the throttle system through an ECU.

(a) One Twizy overview from outside, where the antennas and LiDARs can be observed.



(b) One Twizy overview from inside, where the different devices are depicted.

Fig. 3.9: Renault Twizy 80 instrumented, with different sensors and devices to fulfill CAV functionalities.

For accurate positioning, two different GNSS systems are employed at various stages of testing: the Oxford Technical Solutions (OxTS) xNAV-550, a GNSS system with an integrated IMU supplied by OxTS, and the Duro Inertial, a GNSS system provided by Swift Navigation. Both systems are capable of achieving centimeter-accurate positioning using differential corrections through an NTRIP server.

The vehicles are equipped with two LiDAR systems: a Velodyne VLP-16, primarily used for SLAM applications, and an Ouster OS1 for obstacle detection. For communication with other agents, the Twizys are equipped with a 4G/5G Huawei for internet connection and a Commsignia OBS ITS-4 OBU for V2V and V2I communication.

All internal sensor connections within the vehicles are facilitated through Ethernet, using a NetGear switch with port mirroring capabilities that provide adequate bandwidth to support the aforementioned sensors.

Given the nature of the internal connections and internet access, a Kaspersky IoT Secure Gateway 1000 is installed alongside a host PC (an ASUS ExpertCenter PN64-BB7014MD running Linux 20.04) as part of the IoTAC project. This setup serves as a protective measure against potential cyber-attacks on the vehicles.

3.2.2.2 Irizar Bus

As part of the EU project SHOW, an Irizar 12-m i2eBus was used for conducting servicing tasks within a depot. Similar to the Twizys, this bus is equipped with technology to automate steering, throttle, and braking actions. Figure 3.10 provides a visual representation of this bus, including the distribution of the sensors.

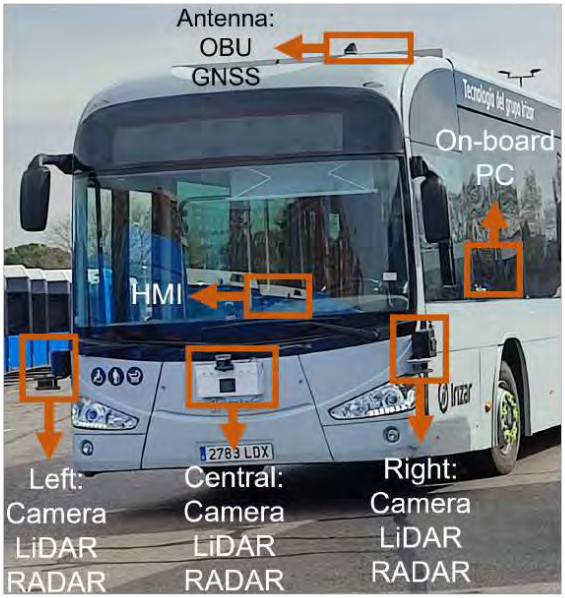


Fig. 3.10: Irizar 12-m i2eBus, with different sensors and antennas installed to achieve CAVs functionalities.

The bus is equipped with three on-board computers. The first, an NVIDIA DRIVE PX2, processes data received from the sensors. The second, an industrial computer NUVO-5002E, interprets the data from the NVIDIA and creates a representation of the environment, identifying objects and events. The third, a Karbon K 700 running on Linux 18.04, operates the remaining driving architecture components.

Regarding the actuator systems, the bus employs electro-mechanical motors attached to the steering wheel and brake pedals. Additionally, an electronic bypass on the acceleration enables signal transmission to the vehicle's ECU.

Localization is accomplished using the OxTS xNAV-550. Environmental information is gathered through a combination of a Continental SSR-208 RADAR, VLP-16 LiDAR, and Sekonix SF3324-10x camera on both the right and left sides. The frontal side is equipped with two cameras the Sekonix SF3324-10x and the Sekonix SF3325-10x, a Continental SSR-300 RADAR, a VLP-32 LIDAR.

For V2X connectivity, the bus employs a Commsignia OBS ITS-4 OBU. All internal sensor connections within the vehicle are facilitated through Ethernet with multiple switches, centralized by a 4G router.

3.2.3 Mixed Environment

The final testing environment integrates both real and simulated platforms, with the simulated platform functioning as a digital twin. The primary objective of this hybrid environment is to facilitate the safe execution of maneuvers while still incorporating variables derived from real-world conditions. Moreover, the utilization of simulators enables reliable visual monitoring of the vehicles. Figure 3.11 depicts the three different methods used to conduct these tests.

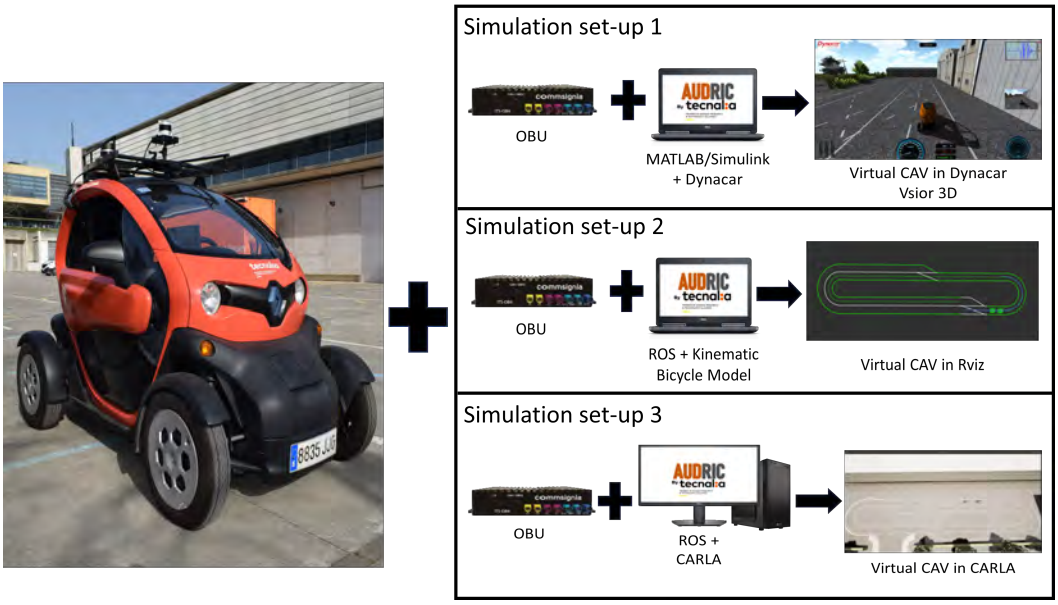


Fig. 3.11: Three mixed scenario configurations used during the Ph.D thesis. Each involving the usage of the Renault Twizy 80, with different simulation set-ups.

Three different set-ups are used for the simulation during the course of this research. The first configuration involved a Commsignia OBS ITS-4 OBU that was connected to a Dell laptop running the driving architecture in Matlab/Simulink in conjunction with the Dynacar simulator. The second set-up also used a Commsignia OBS ITS-4 OBU, but the Dell laptop in this case was running the driving architecture in ROS using a kinematic vehicle model. Finally, the third set-up consisted of a Commsignia OBS ITS-4 OBU connected to a Corsair 100 workstation that ran the driving architecture in ROS and CARLA. Each of these configurations employed the Twizy as the real platform, and the test tracks environments for these simulations were modeled according the scenarios described in the following section.

3.2.4 Test Tracks

The primary test track used in this thesis is the Tecnalia Test Track, situated in the Tecnalia 700 building in Derio, Basque Country. The track encompasses an area of approximately 80 m in length and 10 m in width, with a road width of 3 m. This configuration allows for effective testing of the proposed algorithm on the automated Twizy platform. The track boasts a variety of features, including two roundabouts,

sections designated for lane change, dual-lane paths, and intersections. Additionally, it houses a gatehouse, serving as the location for the infrastructure hardware and the simulation setups. A detailed illustration of these features is presented in Figure 3.12.

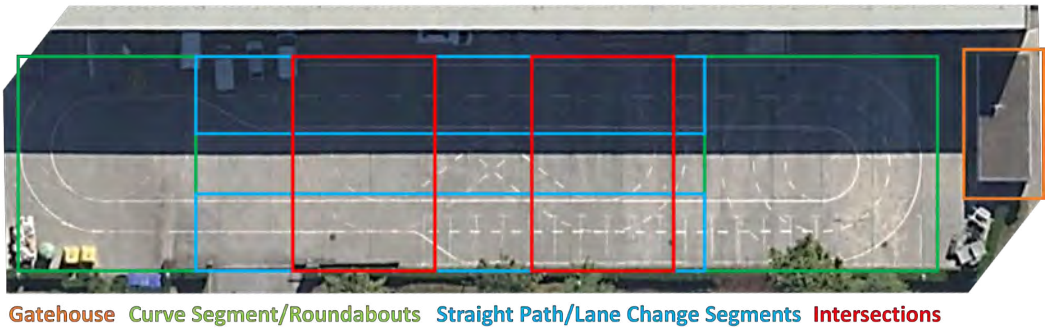


Fig. 3.12: Tecnalia Test Track view from from Google Earth.

The second scenario pertains to an exhibition track, specifically designed to showcase CAVs functionalities during the Go Mobility Expo. This event took place at Ficoba, Irun, in the Basque Country. The exhibition track constitutes a closed circuit, approximately 200 m in length and 6 m in road width. Further visual representation of this scenario can be seen in Figure 3.13.

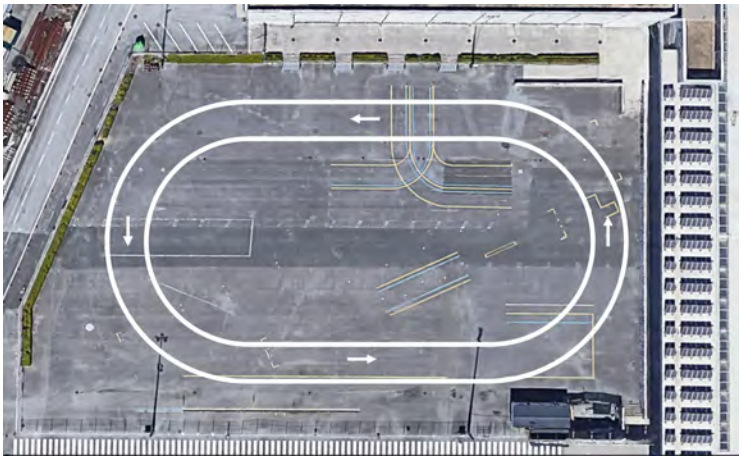


Fig. 3.13: Ficoba Exhibition Track view from Google Earth.

The third testing environment encompasses a depot managed by the Empresa Municipal de Transporte de Madrid (EMT) situated in Carabanchel, Madrid. This particular environment presents a diverse array of circumstances, including the presence of pedestrians, non-automated buses, and various vehicles, as well as an assortment of road segments. A graphical representation of this scenario is provided in Figure 3.14

The fourth and final environment encompasses the vicinity of the Tecnalia 700 building. This setting embodies a comprehensive urban environment complete with diverse road segments and applications related to this Ph.D. thesis, including

roundabouts, pedestrian crossings, and on-ramp merging, among others. It should be noted that this environment was only used for simulation purposes. This scenario is depicted in Figure 3.15.

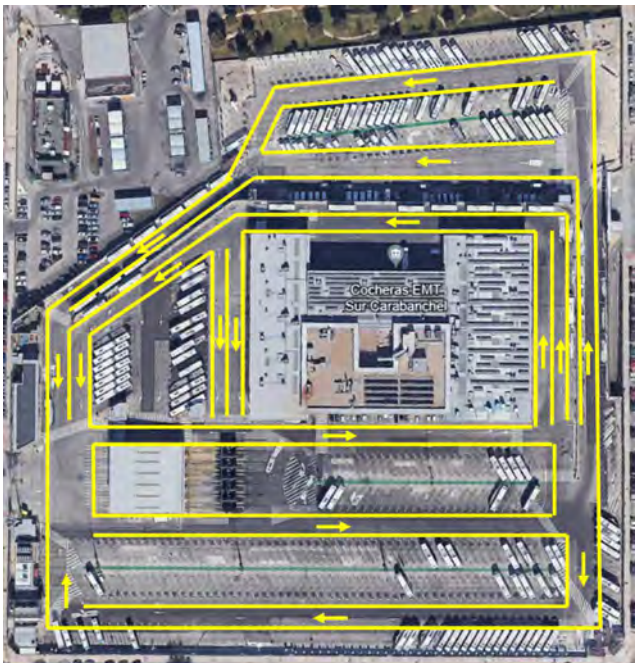


Fig. 3.14: EMT depot view from Google Earth.

Fig. 3.15: Tecnalia surrounding scenario view from Google Earth.

3.3 Summary and Conclusions

This chapter provides a comprehensive overview of the AUDRIC architecture, which has been designed to prioritize security, adaptability, and compatibility with a diverse array of software platforms and vehicles. The architecture's modular task structure enables it to navigate safely under varying traffic conditions, while its

advanced environmental detection and modeling capabilities further enhance its robustness.

In addition to examining the AUDRIC architecture, this chapter also explores the utilization of the SerIoT and IoTAC cyber-secure systems within infrastructure environments. These robust systems provide a reliable framework for real-time monitoring of traffic and IoT network exchanges, thereby ensuring the security of network elements and clients.

Lastly, the chapter delineates the experimental setup employed for testing CCAM functionalities. Notable mentions include the use of software and simulators such as MATLAB/Simulink, ROS, Dynacar, and CARLA. Among these, the combination of ROS and CARLA emerged as the preferred choice due to its superior capacity for simulating a wide variety of complex scenarios. This chapter also mentions the usage of actual vehicles, such as the Renault Twizy 80 and the Irizar i2eBus, in the experimental setup. Additionally, it features a comprehensive depiction of various test tracks employed for conducting the experiments.

The AUDRIC architecture has been instrumental in foundation all the publications of this thesis, including its own presentation paper [198]. Furthermore, it has served as the central architecture for every project involved in this work. With regard to testing methodologies, the mixed environment has proven to be the most effective and safe method for testing, with its first usage in the publication [199].

In the next chapter, the specific algorithms used to carried out the cooperative maneuvers will be presented.

4 Cooperative Maneuver Decision and Control Developments

"Si tu lo deseas puedes volar, Si tu quieres el cielo alcanzar Y las estrellas tocar." Digimon - César Franco

The successful execution of any type of vehicular maneuvers is contingent upon two pivotal stages: decision and control. The decision stage, as reviewed in pre-existing literature, is responsible for regulating the strategic plans for the vehicle, taking into account various surrounding variables, and after that generate a safe and feasible trajectory, aligned with the objective predefined. On the other hand, the control stage is in charge of executing the necessary commands to achieve the trajectory and speed profile.

Numerous research initiatives have been dedicated to ensuring CAVs can execute cooperative maneuvers, in a safe and efficient manner. These studies consider variables such as other vehicles' speed and position, the space available for merging, and the desired final road position. Crucially, it is not only the trajectory that the vehicle must follow that is of importance, but also the speed profile required to reach that position without a collision.

In the domain of cooperative maneuvers, car following has emerged as a pivotal area of research and development, involving a group of vehicles maintaining precise spacing and synchronized movements. By leveraging the collective capabilities of multiple vehicles, car following holds the potential to enhance traffic efficiency, reduce fuel consumption, and minimize congestion on roadways. In this regard, this thesis employs this strategy as a cornerstone for executing other cooperative maneuvers, managing the speed profile that the vehicles must follow when they drive as a platoon.

This chapter begins with an explanation of the control algorithm used for the car following strategy in Section 4.1, encompassing the spacing policy, string stability criteria, and employed technologies (ACC and CACC), all of which are grounded in the findings from the literature review presented in Section 2.2.1.

Following this, the chapter discusses two decision methods for cooperative maneuvers, focusing primarily on merging but not limited to it. Both methods are based

on the application of Bézier curves for trajectory generation. The first approach, termed HYTP as detailed in Section 4.2, employs a static trajectory generation structure, which is enhanced with predictive methodologies to allow for online trajectory modifications. In contrast, the second strategy, RTTP outlined in Section 4.3, uses a more comprehensive decision framework that incorporates extensive information about the map and other agents. This information feeds a numerical optimization technique for real-time trajectory generation. Notably, both strategies integrate the car following controller developed in Section 4.1.

The evolution of the decision strategy was primarily prompted by the observed limitations in the HYTP approach, which was the prevalent methodology within the CCAM group at the beginning of this thesis. These constraints became more apparent when the approach was modified to broaden the spectrum of maneuvers and meet the requirements of demanding scenarios in projects such as SHOW, IoTAC or AUTOEV@L, which primarily involved platoon maneuvers with extend decision-making capabilities.

In order to better comprehend the reason behind the formulation of these distinct methods, a comparison is presented in Table 4.1 from the perspective of the decision module structure delineated in Section 2.1.4, encompassing the global planner, behavioral planner, and local planner. Additionally, the table also lists the platforms on which these methods were tested as well as the EU Project framework within which these methods were developed.

From the table, it can be discerned that the RTTP use a more comprehensive map structure, incorporating a search algorithm that facilitates vehicle navigation from point A to point B, unlike the HYTP, which employs a simple map with descriptive points of the route. In relation to the behavioral planner, the HYTP does not possess such a model per se, but a maneuver planner that allows the vehicle to execute a single maneuver at a time. In contrast, the RTTP use a FSM that facilitates transitions between various states, in this case, maneuvers. Finally, with regard to the local planning, HYTP employs a static trajectory of 4^{th} and 5^{th} order Bézier curves, whereas the RTTP employs an optimization method based on 3^{rd} order Bézier curves to generate trajectories that vary over time. The order of the curve is intentionally reduced to maintain a balance between detail and computational efficiency. This balance is possible due to the extensive map information that the RTTP method possesses.

Nonetheless, the HYTP demonstrated its efficacy by providing a method to test a merging methodology, which was subsequently extrapolated to the RTTP. Additionally, it successfully met the requirements of projects such as ENABLE, Autolib, SerIoT, and the INRIA stays.

Following the discussion on both decision approaches, a fundamental aspect for the execution of cooperative maneuvers lies in the vehicles' capacity to exchange data and articulate their intentions, thereby enabling each vehicle to comprehend

Tab. 4.1: Comparative between HYTP and RTTP

		HYTP	RTTP
Project Framework		- Enable (2018). - Autolib (2019) - INRIA Stays (2020). - SerIoT (2021).	- IoTAC (2023). - AUTOEV@l (2023). - SHOW (2024).
Decision Stage	Global Planner	Simple map with descriptive points that define road structures with no routing algorithm.	OSM map with rich information about the road (e.j. right and left lanes, signals, crossroads, etc) with A* routing algorithm.
	Behavioral Planner	Maneuver planner, with configurations for overtaking, car following, and merging maneuvers by adjusting the MPC inputs.	FSM with capabilities for lane keeping, car following, merging, parking, and overtaking
	Local Planner	Nominal trajectory based on 4^{th} and 5^{th} order Bézier curves.	Real time trajectories based on 3^{rd} order Bézier curves optimization
Platform tested		- MATLAB/Simulink + Dynacar Simulator. - Renault Twizy 80.	- ROS + CARLA Simulator. - Renault Twizy 80. - Irizar I2eBus.
Car Following Strategy		Yes.	Yes.
Negotiation With V2X		No.	Yes.

its role in maneuver execution. Section 4.4 delves into the maneuver negotiation framework deployed for the execution of platoon maneuvers, which can similarly be adapted for merging maneuvers. This encompasses an detailed explanation of the messages structures and the sequential progression of maneuvers.

4.1 Car Following Strategy

Different control approaches have been used for car following maneuvers. From simpler classical control methods to more complex machine learning algorithms. Regardless of the technology or application, one particular technique stands out, the feedforward/feedback with classical control, specifically with PD control. Although it may not yield the optimal performance, this method presents sufficient robustness in its results. Furthermore, it is not only easy to implement, but also capable of demonstrating string stability.

In this work, both ACC and CACC technologies are developed, using a feedforward/feedback with PD control technique. The implemented technologies facilitate a transition from a non-following state to a cooperative following state. The feedforward/feedback strategy is selected for its robustness and simplicity, which enables seamless integration with other strategies to accommodate a broad array of scenarios. Particularly, this strategy enhances the efficiency of the merging scenario.

The virtual vehicle model used is based on the Renault Twizy 80, which led to the creation of a homogeneous platoon of vehicles with identical dynamics. In terms of V2V communication topology, a Predecessor-Only following structure is adopted, wherein only the predecessor's information is used to compute the control algorithm, despite each vehicle transmitting their information to the entire platoon. The adoption of this structure is primarily due to the fact that it facilitates easy manipulation of the string, thereby enabling seamless merge or split operations with minimal complications.

Figure 4.1 illustrates the control architecture of the feedforward/feedback controller for this work. This architecture comprises various components, being $G_{pfi-CACC}(s)$ the groups of blocks comprising the CACC, $G_{pfbi-ACC}(s)$ the group of blocked used for the ACC, and finally the $Gp(s)$ group blocks signifies the transfer function responsible for generating the vehicle's position (Equation 4.2).

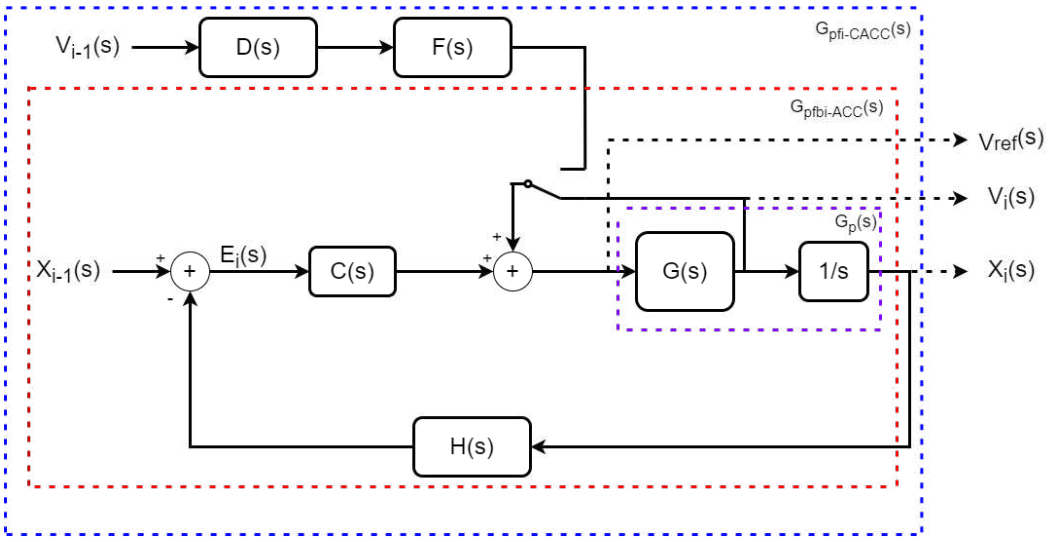


Fig. 4.1: Control structure for ACC and CACC techniques performed in the thesis.

The $G(s)$ block, embodies a second-order function that models the vehicle's response to the low-level control tasked with following the reference speed, V_{ref} (Equation 4.1). To ascertain the values for these transfer functions, $G(s)$ and $Gp(s)$, a series of speed alterations were sequentially introduced within the virtual Renault Twizy 80 simulated in Dynacar, with the objective of observing the platform's response to these changes. This simulator is used specifically due to his accurate representation of the Renault Twizy 80 in virtual environments. Following this procedure, the system identification Integrated Development Environment in MATLAB

is employed to determine the transfer function, and then the controller values. This procedure is followed in accordance to the research of Flores, et Al [104].

$$G(s) = \frac{V_i}{V_{i-1}} = \frac{1.1792}{s^2 + 1.7539s + 1.199} \tag{4.1}$$

$$Gp(s) = \frac{G(s)}{s} = \frac{1.1792}{s * (s^2 + 1.7539s + 1.199)} \tag{4.2}$$

The block denoted as $H(s)$ signifies the selected spacing policy, specifically the Constant Time Gap Policy [200]. This policy is represented by Equation 4.3, where d_{std} refers to a predetermined, fixed standstill distance, h describes the time gap, and $v(t)$ indicates the speed of the ego-vehicle. This relationship can be further elucidated through Equation 4.4.

$$d_{ref}(t) = d_{std} + hv(t) \tag{4.3}$$

$$H(s) = 1 + hs \tag{4.4}$$

The block denoted as $F(s)$ represents the feed-forward filter, as per Equation 4.5. The selection of this element enables an in-depth analysis of the plant and facilitates the acquisition of control variables. This is conducted under the assumption of a homogeneous platoon and absence of communication delay, thereby theoretically ensuring string stability.

$$F(s) = \frac{1}{H(s)} \tag{4.5}$$

The $D(s)$ block symbolizes the delay in communication, as denoted by Equation 4.6. In scenarios limited to simulation, this delay is presumed to be 0. Nevertheless, in real and mixed cases, the delay fluctuates between 40 ms and 100 ms, as dictated by the closed-circuit scenarios applied in this work. To fully comprehend the constraints of the system, analyses examining the system's stability were conducted. These involved altering the delay and noting the smallest possible time gap that can be established while still adhering to the condition of string stability. This analysis, depicted in Figure 4.2, demonstrates that when delays of 100 ms occur, a maximum time gap of 0.6 s can be implemented.

$$D(s) = e^{-\theta * s} \tag{4.6}$$

The $C(s)$ block, responsible for gap regulation, utilizes a PD controller. This controller is known for its effective performance with feed-forward structures [201]. The controller values were derived through a procedure outlined in Flores, et Al [202], which is premised on ensuring robust system response to variations in loop gain via string stability with a reduced time gap.

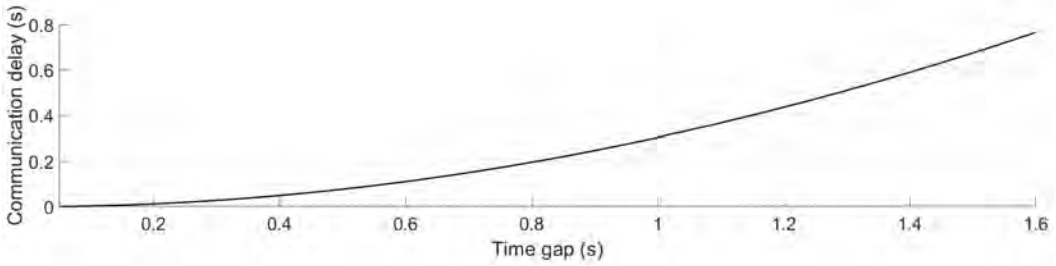


Fig. 4.2: Delay vs minimum time gap analysis.

This methodology includes an examination of the system bandwidth and phase margin over the loop expression $L_{ACC}(s) = G_{pfbi}(s)C(s)H(s)$ for the ACC (blocks within the red dot rectangle in Figure 4.1), and $L_{CACC}(s) = G_{pfi}(s)C(s)H(s)$ for CACC (blocks within the blue dot rectangle in Figure 4.1). The gaincross frequency, w_{gc}, is set to align with the system, and the phase merging, Φ around 45^o.

Upon applying the String stability criteria L_∞, it can be derived the equations to study the string stability for these systems: Equation 4.7 for ACC and 4.8 for CACC.

$$\| \Gamma_{ACC} \|_\infty = \| \frac{X_i}{X_{i-1}} \|_\infty = \| \frac{G_{pfb}(s)C(s)}{1 + G_{pfb}(s)C(s)H(s)} \|_\infty \leq 1; i \geq 2 \tag{4.7}$$

$$\| \Gamma_{CACC} \|_\infty = \| \frac{X_i}{X_{i-1}} \|_\infty = \| \frac{D(s)F(s) + G_{pf}(s)C(s)}{1 + G_{pf}(s)C(s)H(s)} \|_\infty \leq 1; i \geq 2 \tag{4.8}$$

For the control objective, the chosen equation 4.9 employs the inverse of the spacing policy. This selection aims to ensure that the desired loop stability persists, even in the face of minor variations in the plant gain.

$$C(s) = \frac{K_p + K_d s}{1 + hs}; K_p, K_d, h > 0 \tag{4.9}$$

Given that the controller is equipped with two parameters for tuning, two specific control parameters have been established as follows:

- To ensure a desired phase margin of $(45 \pm 1)^o$:

$$arg(L(jw_{gc})) = -\pi + \omega_m \tag{4.10}$$

- To assure that the open-loop gain-cross frequency is equivalent to the system which corresponds to $(1.0950 \pm 0.1) fracrads$:

$$\| L(jw_{gc}) \|_{dB} = 0 \tag{4.11}$$

Equations 4.10 and 4.11 were utilized forming a non-linear equation system. The MATLAB optimization toolbox's *fsolve* algorithm is then employed to ascertain the controller's values. This problem can be construed as a non-convex optimization

algorithm scenario, where potential solutions may be local or sub-optimal. As a result, multiple initial guesses were tested until a solution meeting the pre-established specifications was found. This process led to the determination of the following controller values:

- $K_p = 0.5393$ and $K_d = 0.4103$

Upon the application of this constant to Equation 4.8, and in the context of System 4.2, the Bode plot derived from the system frequency analysis (as depicted in Figure 4.3) is achieved. This illustration demonstrates the fulfillment of the string stability criterion.

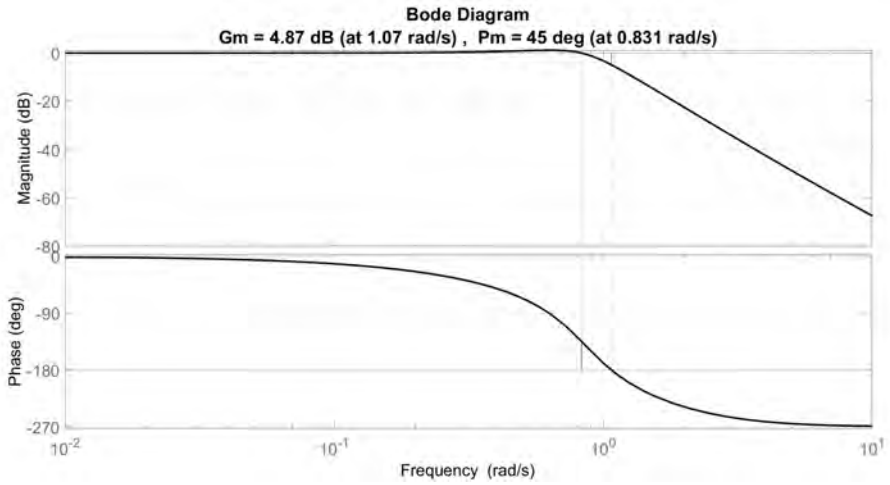


Fig. 4.3: String stability frequency analysis

When considering the operations of gap opening and gap closing, a linear function of the time gap is selected [203]. This function is dependent on the distance between vehicles, denoted as $d(t)$, and the velocity of the ego vehicle, represented by $v(t)$. By having knowledge of the ultimate distance (d_{final}) and the speed (v_{final}), the desired time gap (h_{final}) can be computed (as per Equation 4.12). To achieve this, an acceleration that ensures comfort is applied until the vehicle's distance and the ego vehicle's speed correspond to h_{final}. The acceleration, when increased, allows for a quicker attainment of the time gap.

$$\begin{aligned} h(t) &= \frac{d(t) - d_{std}}{v(t)} \\ h_{final} &= \frac{d_{final} - d_{std}}{v_{final}} \end{aligned} \tag{4.12}$$

The output from this controller is the speed reference ($v_{\text{ref}}(s)$) that the vehicle must follow to execute the car following maneuver. In this context, the proposed car following strategy involves using the speed reference from both technologies, ACC and CACC, to establish a platoon of vehicles instead of relying solely on the speed reference generated by a decision method. Specifically, if the vehicle is joining a platoon or conducting car following without communication, it uses the ACC

speed reference. In contrast, it employs the CACC reference when communication is available, as it provides string stability. However, the latter is heavily reliant on the communication link among vehicles, hence the use of both technologies is desirable to enhance robustness in maneuver execution. The determination of the speed reference used is carried out by the decision stage, where it is decided whether to form a platoon or not. Consequently, the subsequent sections delve into the two decision methods developed and how they are adapted to accommodate the car following controller for achieving maneuvers that involves platoons of vehicles.

4.2 Hybrid Trajectory Planning Method

The HYTP method, which combines trajectory generation via Bézier curves (termed as the nominal trajectory) and a predictive technique based on MPC, is used to determine the future states of vehicles involved in the maneuver and to adjust the trajectory as necessary. This method was specifically developed to address unforeseen scenarios such as lane changes or overtaking maneuvers which was first developed by Lattarulo, et al [144]. When the nominal trajectory encounters a potential obstruction, the ego vehicle evaluates the availability and safety of the opposite lane for overtaking the obstacle. The MPC generates a lateral offset signal to regulate the vehicle's movement from one lane to another, while the longitudinal model concurrently works to prevent any collisions with other participants.

Although originally conceived for lane changes and overtaking scenarios, the HYTP method is not exclusively limited to such use cases. In this thesis, this approach is extended to address other scenarios such as merging, focusing on lane and roundabout merging. Despite the lack of explicit consideration for on-ramp merging, the approach may still be deemed a viable solution for such circumstances, given the range of factors it accounts for.

Similar to the overtaking maneuver, merging at a roundabout requires the vehicle to avoid potential obstacles that could interfere the preset trajectory. This is achieved through longitudinal actions for roundabout merging, and a combination of lateral and longitudinal actions for overtaking. It is noteworthy that trajectory generation on roundabouts requires additional considerations compared to a straight path, such as the angles of entrance and exit, and the radius. These considerations become increasingly apparent when cooperation among vehicles is incorporated. For instance, in the case of a roundabout, vehicles must adjust their speed to avoid stopping at the entrance, given that vehicles inside the roundabout have priority. This set of rules changes in overtaking scenarios, where the negotiation process differs.

In the case of lane merging, this maneuver incorporates not only lateral but also longitudinal actions. Laterally, it behaves similarly to a lane change maneuver. However, before this process starts, the vehicle adjusts its position as necessary by reducing or augmenting its speed. When cooperation is introduced, vehicles can negotiate their positions to proceed in the safest and most optimal manner.

Figure 4.4 presents a block diagram of the HYTP approach. The diagram receives two inputs: 1) data from other vehicles (speed, position, acceleration, yaw, etc.), and 2) data from the ego vehicle (speed, position, acceleration, yaw, etc.). The information from the ego vehicle is directed to the Nominal Trajectory Calculator block, which is fundamentally grounded in Bézier curves. Additionally, this approach incorporates a maneuver planner component which receives input from both the ego vehicle and other agents. This component houses a MPC system that generates speed and lateral references to adjust the nominal trajectory in response to any eventualities.

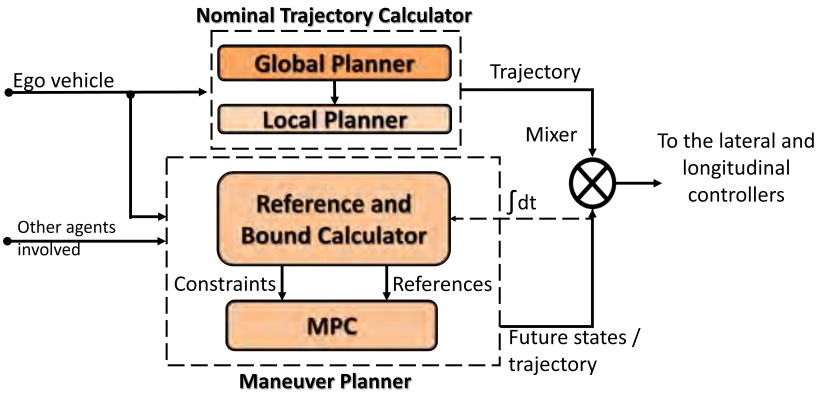


Fig. 4.4: HYTP block diagram. Consisting on two main modules; Nominal Trajectory Calculator, and the Maneuver Planner.

4.2.1 Nominal Trajectory Calculator

The methodology for generating the nominal trajectory was established within the framework developed by Lattarulo et al. [204], wherein a variety of road components - encompassing intersections, roundabouts, straight lines, and lane changes - were characterized by integrating Bézier curves in segments along lines and arcs. These curves exhibit geometric (G^n) and curvature continuity (C^n), and can be defined by a specific mathematical expression:

$$\begin{aligned} B\left(t|b, P_0, ..., Pn\right) &= \sum_{i=0}^{n} b_i P_i, \\ b_i &= \binom{n}{i} t^i \left(1-t\right)^{n-i} \end{aligned} \tag{4.13}$$

Where $\{b_i \in \mathbb{R}\}$ is the Bernstein polynomial, $\{P_i \in \mathbb{R}^2\}$ are the control points, $\{n \in \mathbb{N}^+\}$ is the polynomial order, and $\{t \in \mathbb{R}, t = [0, 1]\}$ is the parameter for curve construction. An example of a Bézier curve can be seen in Fig. 4.5, where the following properties are observed:

- The initial point aligns with the first control point P_0, while the terminal point aligns with P_n. In this research, the maximum value of n is confined to 5, a

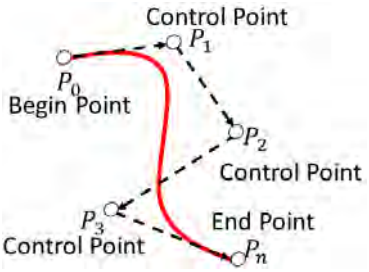


Fig. 4.5: Example of a Bézier curve, indicating how the curve is plotted on the basis of the control points.

parameter chosen based on its optimal computational efficiency and superior results.

- The direction vectors at the start and end of the curve are determined by $\overrightarrow{P_0P_1}$ and $\overrightarrow{P_{n-1}P_n}$ respectively
- The curve is situated within a convex hull formed by the control points.
- Bézier curves exhibits geometric (G^n) and curvature (C^n) continuity, a property that can be maintained when two different curves are jointed
- Bézier curves are symmetric.

Within the scope of this framework, a two-stage planning process is executed. The initial stage, serving as a global planner, and a subsequent stage focusing on trajectory planning, wherein the Bézier control points are outlined to generate a smooth trajectory

4.2.1.1 Global and Local Planner

The global planner employs a map, also known as Simple Map comprised of a basic route which contains description points that defines common road structures, and thus simplifying the local planning approach. Once the first definition of the control points is obtained for each road segment, a smooth and continuous curvature trajectory is generated with different order Bézier curves. For the context of this research, the most pertinent road structures are lane change and roundabout, detailed below.

In the context of **lane changes**, the proposed approach contemplates segments of straight paths during the maneuver as first definition in the global planner. Equation 4.14 delineates the relationship between these two straight paths, with $\vec{u_b}$ and $\vec{u_a}$ representing the unit vectors of each path, respectively.

$$\begin{aligned} \vec{u_b} &= -\vec{u_a} \\ \frac{P_b - P_{lc}}{||P_b - P_{lc}||} &= -\frac{P_a - P_{lc}}{||P_a - P_{lc}||} \end{aligned} \tag{4.14}$$

To accommodate the maneuver in local planning, a new point, P'_{lc}, is necessary. This is informed by the relationship depicted in Equation 4.15, wherein w represents

the road width and $\vec{u_a}$ signifies the unitary vector as per Equation 4.14. The symbol $\pm$ is contingent upon the direction of the lane change being performed; it takes a negative value when the lane change is executed to the right and a positive value when done to the left.

$$P'_{lc} = P_{lc} + w \begin{bmatrix} \cos(\arctan(\frac{u_{ay}}{u_{ax}} \pm \frac{pi}{2}) \\ \sin(\arctan(\frac{u_{ay}}{u_{ax}} \pm \frac{pi}{2}) \end{bmatrix} \tag{4.15}$$

Figure 4.6 illustrates the schematic representation of lane change planning, detailing the positioning of control points and the trajectory's formation. The following considerations have been taken into account during this process:

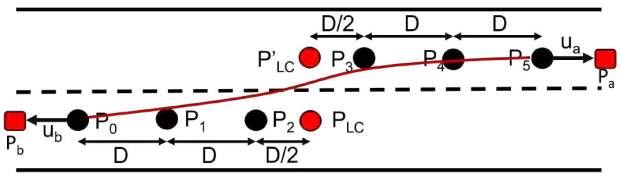


Fig. 4.6: Simple Map and trajectory planning on lane change, with the corresponding Bézier control points.

- Three control points of the Bézier curve (designated as $P0$ to $P2$) are strategically positioned on a singular line. This alignment guarantees that the curve starts with zero curvature, thus representing a straight path at the beginning of the curvature. This same criteria is applied to the terminal section of the trajectory, denoted by the control points $P3$ to $P5$.
- A Bézier curve of the 5^{th} order has been chosen for this work. This choice is justified by its ability to meet the aforementioned criteria, while also offering a greater degree of flexibility than a curve of the 4^{th} order.
- The distance between control points, denoted as D, is governed by the relation $D = ||P_{n-1} - P_n||_{u_a}$. The value of D is subject to an upper limit, specified by the variable w.

In the context of **roundabout** navigation, the approach simplifies the roundabout as a circle for the global planning. This approximation is derived from the roundabout's center point (P_r), its radius (R), and the entrance (a_i) and exit (a_o) angles. The coordinates for the entrance and exit points used for the local planning of the roundabout are subsequently defined based on these parameters as shown in Equation 4.16.

$$
\begin{aligned}
P_{entr} &= P_{cent} + R \begin{bmatrix} \cos(\arctan(\frac{u_{by}}{u_{bx}} \pm a_i) \\ \sin(\arctan(\frac{u_{by}}{u_{bx}} \pm a_i) \end{bmatrix} \\
P_{exit} &= P_{cent} + R \begin{bmatrix} \cos(\arctan(\frac{u_{ay}}{u_{ax}} \mp a_o) \\ \sin(\arctan(\frac{u_{ay}}{u_{ax}} \mp a_o) \end{bmatrix}
\end{aligned}
\tag{4.16}
$$

The points P_{entr} and P_{exit} denote the coordinates for the entry and exit points of the roundabout, respectively. The unitary vectors $\vec{u_b}$ and $\vec{u_a}$, as defined in Equation 4.14, are used in these calculations, substituting the P_i point with the P_{entr} and P_{exit} points. In these equations, the upper sign is employed when traffic is counter clockwise whereas the lower sign is used for clockwise traffic. Figure 4.7 provides a visual representation of the roundabout planning, indicating the placement of the control points and the trajectory's construction. The following considerations have been made in this context:

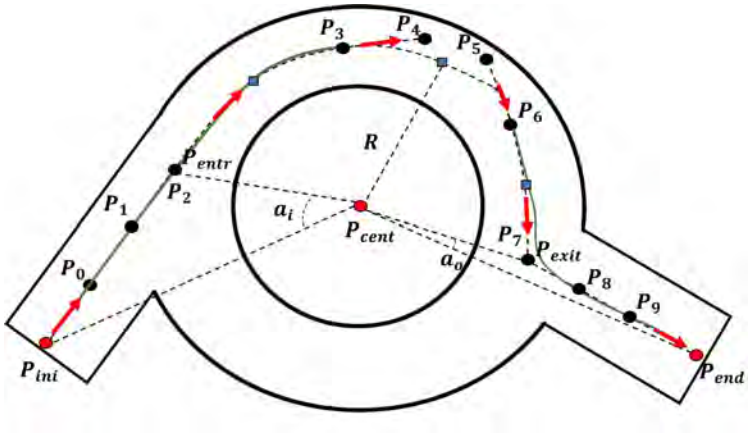


Fig. 4.7: Simple Map and trajectory planning on roundabout of one lane, with the corresponding Bézier control points.

- The curvature at both the entrance and exit must be zero, indicating a straight path. As such, two curves with three col-linear control points each, are strategically designed for both segments. For the entrance, these are designated as P_0,P_1, and P_2/P_{entr} for the entrance, and for the exit, P_7/P_{exit}, P_8, and P_9.
- The segments generated at both the entry and exit of the roundabout must be designed in such a way that the curvature of their inner parts corresponds to the inverse of the radius of the roundabout.
- The angle at the point where Bézier curves converge must be equal to the angle of the circular arc.
- For ensuring both geometric and curvature continuity, two points are strategically chosen to construct a tangent segment to the circle. Specifically, for the

entrance, points P_3 and P_4 are selected, while points P_5 and P_6 are chosen for the exit. These pairs of points are separated by a distance, denoted as D, as illustrated in Equation 4.17. In this equation, $\vec{u_{p3}}$ represents the tangent vector at point P_3, and K_r is indicative of the curvature. In the scenario of the exit, points P_2, P_3, and P_4 are replaced by P_7, P_6, and P_5, respectively.

$$D_{p_4} = \sqrt{\frac{3}{4}\frac{\parallel \vec{u_{p3}} \times (P_2 - P_3) \parallel}{K_r}} \tag{4.17}$$

- A Bézier curve of the 4^{th} order is chosen as it fulfills the previously established criteria.

4.2.2 Maneuver Planner

In the HYTP approach, the nominal trajectory initially described exhibits a static property, indicating its inflexible against varying circumstances. To address this limitation, a maneuver planner is introduced, functioning like to a behavioral planner by altering the vehicle's behavior in response to specific events. One constraint of this module is its fixed nature, set to perform a specific maneuver for each test, such as overtaking or merging. The module consists of two sub-modules: MPC Reference and Bound Calculator, and the MPC itself.

The **MPC Reference and Bound Calculator** use past iteration information (lateral acceleration and longitudinal jerk) and the current state of the vehicles. These are integrated into the MPC block to obtain future states (acceleration, speed and position in the longitudinal case, and distance and speed in the lateral case), which are used to propagate the ego vehicle's and other agent's position over time to check for collisions. Simultaneously, this information, derived from the integration chain, is utilized to adjust the references generated by the nominal trajectory block.

The **MPC** is designed in accordance with the work outlined in [144]. This work employs a simplified linear model, specifically a point mass model, which features decoupled lateral and longitudinal dynamics. The primary motivation behind the selection of this model is its inherent simplicity, which facilitates the discovery of optimal solutions swiftly. The lateral model in this framework is fundamentally based on a double integrator of the lateral acceleration (a_{lat}) component, as formulated in Equation 4.18.

$$d_{lat} = \int\int a_{lat}(t)dt^2 \tag{4.18}$$

This model can be explicated as a system of linear differential equations, denoted as Equation 4.19. Within this system, the state variables comprise the lateral offset (d_{lat}) and its rate of change (v_{lat}), while the control input is represented by the lateral acceleration(a_{lat}).

$$\begin{bmatrix} \dot{d_{lat}} \\ \dot{v_{lat}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d_{lat} \\ v_{lat} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} a_{lat} \tag{4.19}$$

The lateral constraints are defined by the given inequalities, as presented in Equation 4.20.

$$\begin{aligned} -\frac{1}{2}|Road_w| + \frac{1}{2}Veh_w \leq d_{lat} \leq \frac{3}{2}|Road_w| - \frac{1}{2}Veh_w \\ -|v_{max}|| \leq v_{lat} \leq ||v_{max}|| \\ -|a_{min}|| \leq a_{lat} \leq ||a_{max}|| \end{aligned} \tag{4.20}$$

The distance (d_{lat}) is influenced by dynamic constraints, determined by the width of the vehicle (Veh_w) and the road ($Road_w$). Meanwhile, both acceleration (a_{lat}) and speed (v_{lat}) are subject to static constraints, which are defined by the maximum allowable change in the steering wheel and the actuator delay.

The longitudinal model is fundamentally constructed on a triple integrator chain of the longitudinal jerk component, denoted as J_{lon} (refer to Equation 4.21).

$$d_{lon} = \int\int\int J_{lon}(t)dt^3 \tag{4.21}$$

The model can be conceptualized as a system of linear differential equations, incorporating the longitudinal distance (d_{lon}), speed (v_{lon}), and acceleration (a_{lon}) as state variables. Notably, the jerk (J_{lon}) is integrated as the control input, culminating in the subsequent state-space representation (refer to Equation 4.22).

$$\begin{bmatrix} \dot{d_{lon}} \\ \dot{v_{lon}} \\ \dot{a_{lon}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_{lon} \\ v_{lon} \\ a_{lon} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} J_{lon} \tag{4.22}$$

The constraints pertaining to the variables are outlined in accordance with Equation 4.23

$$\begin{aligned} 0 \leq d_{lon} \leq d_{veh_{front}} \\ 0 \leq v_{lon} \leq v_{limit} \\ -|a_{min}| \leq a_{lon} \leq |a_{max}| \\ -|J_{min}| \leq J_{lon} \leq |J_{max}| \end{aligned} \tag{4.23}$$

The relative distance between vehicles, during execution time, possesses varying constraints that range from zero to the maximum pre-collision distance (denoted as $d_{veh_{lon}}$). The speed maintains a lower limit of zero and an upper limit set by the speed limit (notated as v_{limit}). The acceleration is constrained by the vehicle's

maximum permissible acceleration and deceleration values. Lastly, the jerk's upper and lower boundary values are determined in accordance with passenger comfort [205].

The problem is solved using a Quadratic Problem (QP) formulation, which minimizes the cost function $J(x(t), u(t))$, as denoted in Equation4.24. This cost function is computed based on the discrepancy between the longitudinal speed of the ego vehicle ((v_{lon}) and the reference speed generated by the MPC (v_{reflon}). Additionally, it takes into account the difference between the lateral offset (d_{lat}) and the lateral reference produced by the MPC (d_{reflon}). The cost function is subject to the constraints stipulated in Equations 4.20 and 4.23, and the weighting functions L and M.

$$\begin{aligned} \Phi(x(\cdot), u(\cdot)) = min\{ \int_{t_0}^{T} L(J(x(t), u(t)))dt \\ + M(J(x(T), u(T)))\} \\ J(x(t), u(t)) = (d_{ref_{lat}} - d_{lat})^2 + (v_{ref_{lon}} - v_{lon})^2 \end{aligned} \tag{4.24}$$

For this thesis, the MPC constrains are based on the vehicle characteristics employed, that corresponds to the Twizy:

Tab. 4.2: Vehicles characteristics and MPC constrains

	Property	Value
Vehicles Dimensions	Width	2.40 m
	Length	1.30 m
MPC Constrains	Real Vehicle Max. Speed	15 km/s
	Virtual Vehicle Max. Speed	10 km/s
	Vehicles Max. Acceleration	1 m/s^2
	Vehicles Max. Deceleration	3.15 m/s^2
	Max. Jerk	2.5 m/s^3
	Min. Jerk	-2.5 m/s^3
	N^o of Samples	10

The HYTP approach, while it provides a robust focus on avoiding collisions, lacks capabilities in the area of traffic throughput, particularly in the longitudinal domain. Its primary function is to prevent collisions with the vehicle in front, and it is only used to increase speed during overtaking maneuvers. To address this limitation, it is suggested to integrate this method with a car following technology such as CACC to offer additional flexibility in the longitudinal domain. This integration is further elaborated in the following section.

4.2.3 Speed Planning For Cooperative Maneuvers

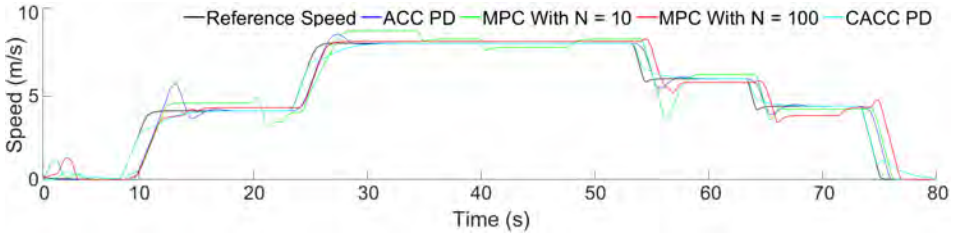
To better understand how the HYTP approach can be improved with a car following control strategy, the Table 4.3 illustrates the advantages and disadvantages of the HYTP approach and the car following control strategy described in Section 4.1, more specifically the CACC. It can be observe that these methods are complementary, enabling both car following and lane change maneuvers without an increase in computational cost.

Tab. 4.3: Advantages and disadvantages of car following and HYTP.

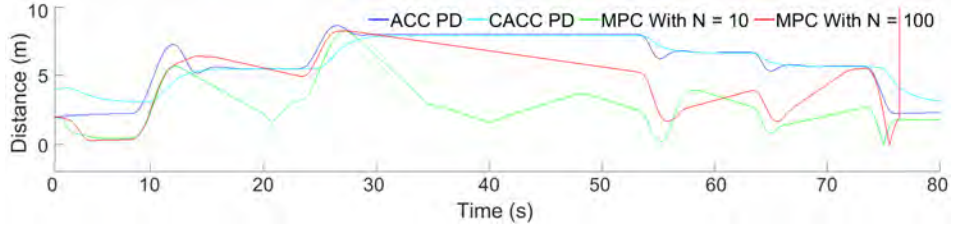
	Advantages	**Disadvantages**
Car following control strategy	Allows short gaps between vehicles	No actions/ usages in the lateral component
	Ease in handling the gap for cut-in, cut-out maneuvers	
	Easy implementation	Difficult lateral displacement
	Low computational cost	
	String stability guaranteed	
HYTP	Actions in both longitudinal and lateral components	Average to high computational cost
	Allows lateral maneuvers such as lane change	
	Collision avoidance through projections of future states	No short gaps allowed
	Passenger comfort guaranteed	
	Easy to medium implementation	

This thesis's findings, pertaining to the longitudinal domain, emphasize that the HYTP approach is more inclined towards ensuring safety and preventing collisions. This is a deviation from traffic throughput, as the approach generates a speed profile to maintain a safe distance ahead of the preceding vehicle, rather than closely following it and maintaining a shorter distance. Consequently, within the context of car following maneuvers, the MPC of the HYTP method tends to be outperformed by other car following algorithms, such as the traditional Feed-back/Feed-forward PD controller. Evidence of this under-performance is provided in Figure 4.8, which reveals that the MPC fails to accurately track the reference speed, unlike the feed-back/feed-forward PD controller in both ACC and CACC modes. The primary reason for this discrepancy lies in the precision obtained from the number of samples used.

In an attempt to enhance the performance of the MPC, the number of samples was increased to 100, each with a sample time of 0.05 s. This increment in predictions resulted in improved measurement precision, which can be corroborated by the



(a) Speed response.

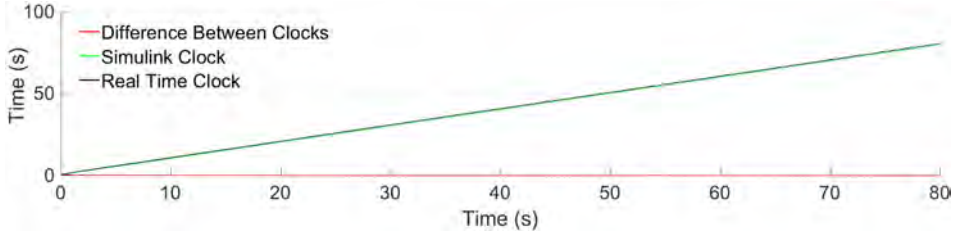


(b) Distance between vehicles.

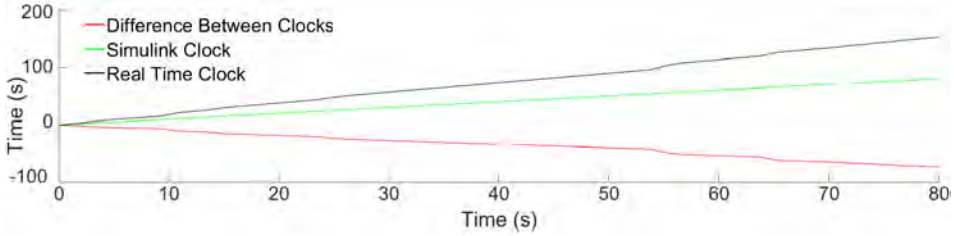
Fig. 4.8: Speed and position response comparative when different controller are used..

alignment of the MPC with the reference as depicted in Figure 4.8. However, this enhancement comes at the cost of increased computational load, thereby potentially compromising the system's real-time functionality.

To illustrate the system's performance, Figure 4.9 contrasts a Simulink clock with a real-time clock. When the MPC uses 10 samples, the two clocks move in sync. However, when the MPC employs 100 samples, as shown in Figure 4.9b, the gap between the clocks progressively widens, suggesting that the system operates at a slower pace.



(a) Simulink clock and real time clock with MPC (Nº samples = 10).



(b) Simulink clock and real time clock with MPC (Nº samples = 100).

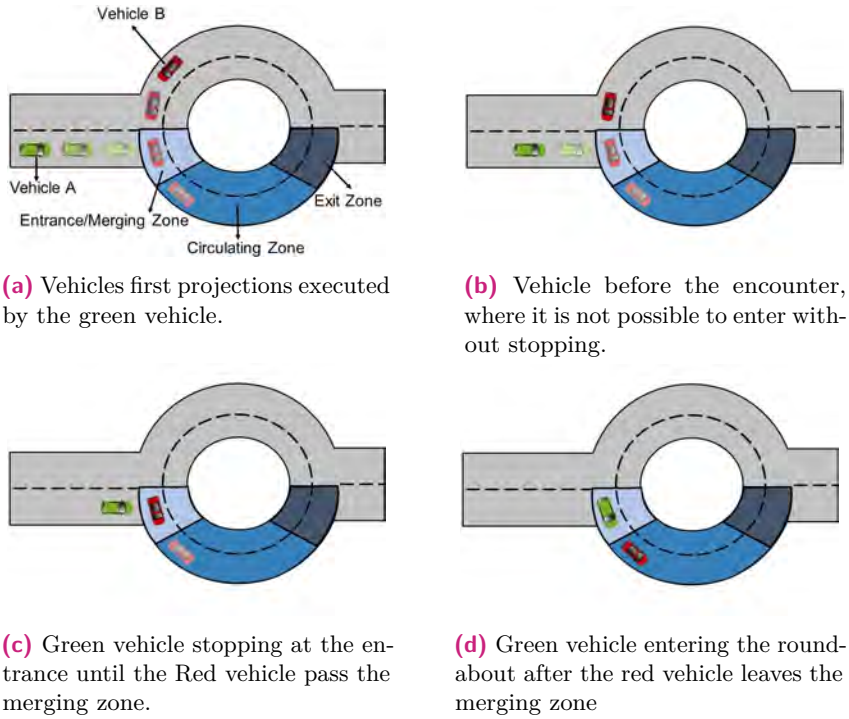
Fig. 4.9: Comparative between the Simulink clock and the real time clock with different MPC sample configurations.

To ensure real-time performance, the feedback/feedforward controller is assigned to manage the car following maneuver generating speed reference. Concurrently,

the MPC within the HYTP approach, operating with a limited number of samples, is specifically designated to handle the lane change maneuver generating lateral references. This configuration facilitates the execution of additional maneuvers involving platoons of vehicles, including but not limited to, platoon merging.

Once the task distribution for both strategies is established to facilitate platoon-related maneuvers, an explanation follows on how these strategies complement each other to achieve the desired outcomes. In scenarios where CAVs lack cooperative capabilities, the adjustment of the ego vehicle's position is achieved through the implementation of a virtual platoon. This involves projecting the position of the leading vehicle on the main road onto the current lane, initiating a virtual follow-the-leader scenario. This process dynamically accommodates changes in the speed of the leading vehicle during the adjustment phase. Additionally, the positions of the ego vehicle and other participants are projected onto their respective lanes using timestamps from MPC and a kinematic vehicle model, facilitating the assessment of potential collisions.

In relation to the specific case of roundabout merging scenarios, as depicted in Figure 4.10, if the ego vehicle cannot adjust its position to join the roundabout without stopping, it yields at the entry until other vehicles on the roundabout have passed the merging zone. Only then, does the ego vehicle enter the roundabout.



(a) Vehicles first projections executed by the green vehicle.

(b) Vehicle before the encounter, where it is not possible to enter without stopping.

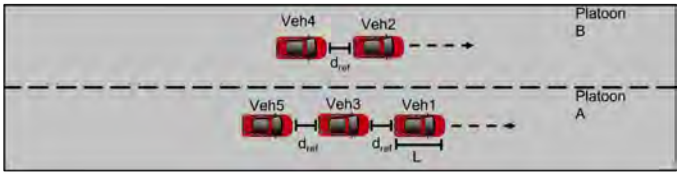
(c) Green vehicle stopping at the entrance until the Red vehicle pass the merging zone.

(d) Green vehicle entering the roundabout after the red vehicle leaves the merging zone

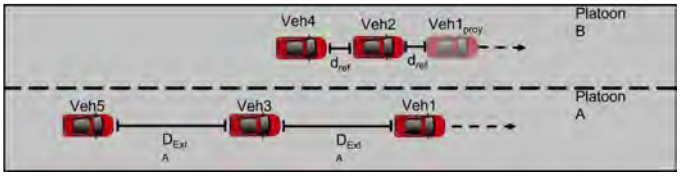
Fig. 4.10: Images sequence of the merging at roundabout with vehicles projections, where the green vehicle generates projections of both vehicles in order to enter the roundabout.

In the context of platoon presence, Figure 4.11 delineates the sequence for lane merging. This case comprises two distinct processes. The first entails vehicles situated in the "main" lane, referred to as Platoon A. The second pertains to the

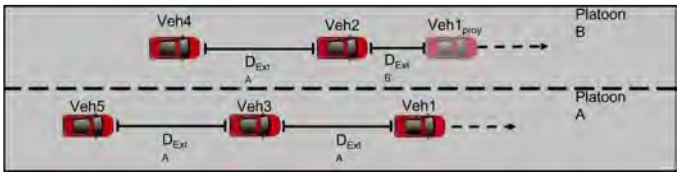
vehicles intending to join, referred to as Platoon B. Vehicles in Platoon A create the necessary space to facilitate the merging of Platoon B vehicles through a gap opening operation. This is in accordance with Equation 4.25, wherein *Length* represents the length of the vehicles, and d_{std} symbolizes the standstill distance utilized in the car following algorithm.



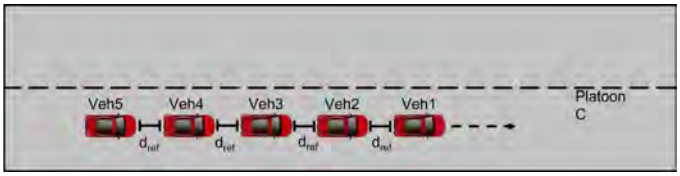
(a) Platoon A and B before the lane merging maneuver execution.



(b) Platoon B following the projection of the Platoon A leader in their lane, whereas the Platoon A opened up the space.



(c) Platoon B positioned to initiate the lane changing into the spaces opened by the Platoon A.



(d) Platoon C formed by the combination of Platoon A and B.

Fig. 4.11: Illustration of platoon merging maneuver as sequence of steps, from top to bottom picture. Where the Platoon B seeks to join Platoon A to form Platoon C.

$$D_{ExtA} = 2 * d_{std} + Length \tag{4.25}$$

Meanwhile, the leader of Platoon B starts the process by following the projection of the Platoon A leader. The gap is opened according to Equation 4.26, which incorporates D_{ExtB} to represent the distance between vehicle i and $i-1$ in Platoon B. $Pos_{x_{Proy}}$ and $Pos_{y_{Proy}}$ denote the position of the Platoon A front vehicle projected by a $D_{ExtA}/2$ distance into the Platoon B lane. Finally, $Pos_{x_{i-1}}$ and $Pos_{y_{i-1}}$ represent the position of the $i-1$ vehicle in Platoon B.

$$D_{ExtB} = \sqrt{(Pos_{x_{i-1}} - Pos_{x_{Proy}})^2 + (Pos_{y_{i-1}} - Pos_{y_{Proy}})^2} \tag{4.26}$$

Upon successful adjustment and stabilization of each vehicle's position, Platoon B initiates the lane change process while maintaining their current speed. Upon completion of the merge, Platoon B joins Platoon A to form Platoon C. Considering Equations 4.25 and 4.26, the vehicles are positioned to maintain the distance d_{ref}, thereby allowing the reference to be switched back to the one generated by the car following algorithm without causing disturbances in Platoon C.

4.3 Real-Time Trajectory Planning Method

As the complexity of CCAM technologies increases, the demand for more refined map data and an enhanced behavioral planner becomes apparent. These improvements would allow the vehicle to make real-time decisions and adjust its trajectory planning accordingly. It has been identified that the existing HYTP falls short in these areas, leading to the exploration of an alternative approach, known as Real-Time Trajectory Planning.

This method seeks to generate real-time trajectories through an optimization-based solution, which contemplates the entire solution-space with a minor compromise on computational time. It stands in contrast to sampling-based methods, which necessitate significant computational resources to generate a discrete set of pre-computed motion primitives.

Given that the method is contingent on the geometry of the road, it employs Bézier curves to construct a secure trajectory that aligns with a collision-free corridor, termed as the drivable space. This planning process leverages precise map information to initially create a static representation of the drivable space that can drive the vehicle from its starting point to its destination. Subsequently, a decision-making algorithm adjusts this drivable space in accordance with dynamic data, culminating in the generation of an optimized trajectory.

4.3.1 RTTP: Global Planner

The global planner makes use of highly precise maps that are structured in the XML-based OSM data format, adhering to a condensed version of the Lanelet2 standard [206]. This format is compatible with multiple publicly available editors and viewers, with the JOSM[1] editor being used in this research. The method assumes that all elements on the map can predominantly be delineated by a projection onto a flat ground plane, a requirement that is typically met for all elements close to a road. While height information is significant for determining the height profile of a road, its implementation is beyond the scope of this application.

In building the map, several elements are utilized:

- **Points**: These are the fundamental units of the map, typically constituting parts of line strings. Each point is characterized by its three-dimensional

[1] **Webpage**: JOSM → https://josm.openstreetmap.de/

position in metric coordinates, and uniquely, they are the only primitives with position information.

- **Line Strings**: These are ordered arrays consisting of two or more points, between which linear interpolation occurs. They are instrumental in representing the shape of map elements such as road markings, curbs, facades, and fences. Some line strings are virtual, forming implicit borders of lanes. Line strings, due to their capacity to describe any one-dimensional form through high discretizations, were selected as the choice form of representation.
- **Lanelets**: These define atomic sections of the map where directed motion occurs. Examples include normal lanes, pedestrian crossings, and rails. 'Atomic' refers to the consistent traffic rules within a lanelet and the unchanging topological relationships with other lanelets. Each lanelet is delineated by a single line string on the left and right borders. It may also encompass several regulatory elements that express applicable traffic rules. Lanelets can intersect or overlap, and the type of border determines the possibility of lane changes to an adjacent lanelet. Consecutive lanelets share the endpoints of the left and right borders. Movement in the opposite direction may be permitted within a lanelet, causing the left border to become the right border and vice versa.
- **Regulatory elements**: These define traffic regulations such as speed limits, priority rules, or traffic lights. A regulatory element always corresponds to one or more lanelets or areas to which they apply.

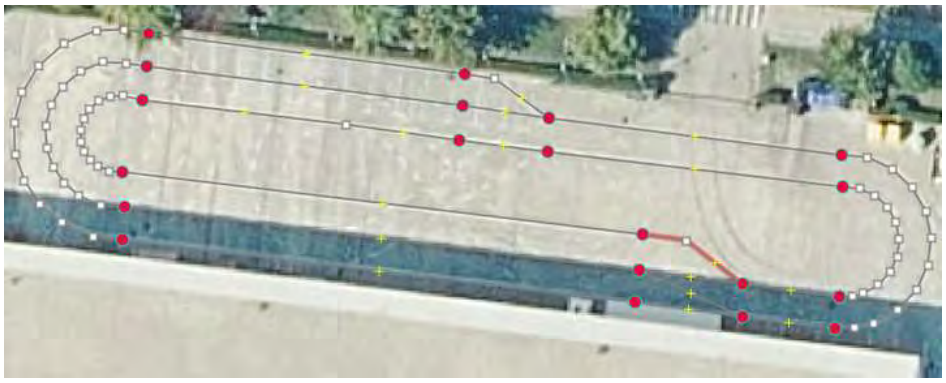
The map employs a locally fixed reference system, such as ETRS89 in Europe, which uses lossless geographic coordinates (latitude/longitude) to bolster robustness against continental drift, while maintaining immutability during map reading. Upon offline construction, the map is stored by the algorithm, which converts geographic coordinates into a local metric coordinate system (in this case, UTM), facilitating efficient calculations. The map's storage is designed for easy access, allowing other internal algorithms to readily retrieve the information.

Subsequent to map storage, the search algorithm, based on the A* algorithm, identifies the shortest path between points. For this function, the astar-algorithm[2] library in C++ is used due to its high performance in real-time applications and ease of integration.

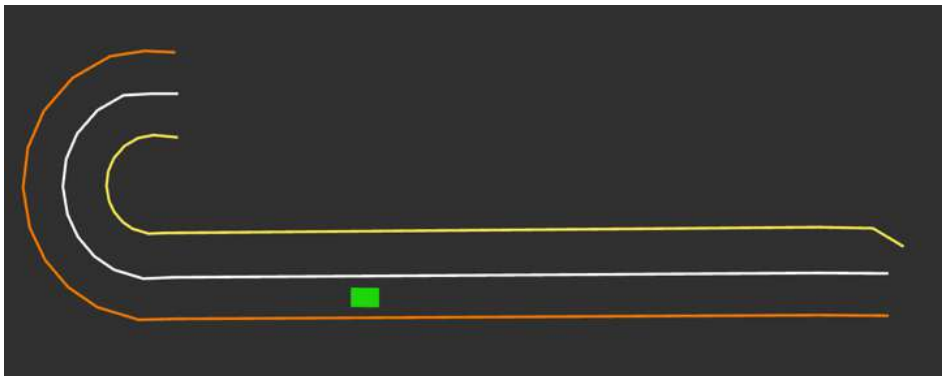
Once the necessary lanes for reaching the predetermined destination are identified, the vehicle's current position is used to establish its location on the map. Only those segments of the map that are necessary for reaching the destination within a defined horizon view are considered, constrained by a configurable maximum distance to reduce the computational load. This constraint is employed to delineate a drivable space within which the vehicle can maneuver without imposing excessive computational requirements. This initial representation of the drivable space is static, taking into account only the information from the map (e.g., lane structures,

[2] **Webpage**: astar-algorithm → https://github.com/justinhj/astar-algorithm-cpp

regulatory elements, etc.). An example can be observed in Figure 4.12a, which displays a representation of the Tecnalia test track map in the JOSM application. Meanwhile, Figure 4.12b illustrates the drivable space generated from this map, where the yellow line designates the left margin of the drivable space, the orange line signifies the right side, the white line demarcates the lane division, and the green rectangle denotes the vehicle.



(a) Example of the Tecnalia test track representation in the JOSM application.



(b) Drivable space representation acording the position of the vehicle.

Fig. 4.12: Views of the Tecnalia Test track according the JOSM application and the drivable space.

The drivable space representation is processed by a subsequent algorithm that incorporates dynamic data, such as the presence of other vehicles, traffic lights, pedestrians, and route challenges. Consequently, a second iteration of the drivable space is generated, which takes these factors into account. This stage, defined as behavioral planner is explained in the subsequent section.

4.3.2 RTTP: Behavioral Planner

Given the limitations of the HYTP maneuver planner, various strategies were examined in the SoA. Techniques based on optimization, RL, and FSMs stood out as the most used. However, it's important to note that RL and optimization techniques could be computationally demanding and complex to implement within this phase.

In contrast, FSMs offer a systematic and structured approach to decision-making within the context of CCAM, rendering them a compelling alternative. FSMs furnish

a lucid and concise depiction of the system's behavior. Each state corresponds to a specific system condition, while the transitions dictate the conditions under which the system transitions from one state to another. This clear representation simplifies the understanding and analysis of the system's behavior. FSMs adhere to predefined states and transitions, thereby ensuring deterministic behavior which is particularly advantageous in safety-critical scenarios where predictability and repeatability of actions are paramount.

FSM decision-making methodologies have been employed to address CAV maneuvers, such as overtaking [207] and highway driving [208]. This includes the incorporation of lane changes and ACC, resulting in promising outcomes. However, these FSM approaches are limited in scope, lacking in cooperative and urban environments.

In the specific application of FSMs for solving cooperative maneuvers, a few studies, such as [209], have used it for planning a cooperative roundabout merging. However, this approach is confined to that specific case and lacks a comprehensive description of the cooperation aspect of the maneuver. Other studies, such as [210] or [211], propose a more complex FSM that can handle platoon lane merging operations. Yet, these approaches present scalability issues, as they do not contemplate maneuvers like roundabout merging or overtaking, nor do they consider urban environments.

In summary, most of the current approaches are focused on a specific maneuver or case, neglecting other important maneuvers or environments. The proposed FSM in this research is designed to address multiple scenarios in both cooperative and non-cooperative settings, with a primary focus on urban environments.

The FSM adheres to the relation delineated in Equation 4.27. In this equation, Hs represents the set of states corresponding to the maneuvers, whereas $\sum$ represents the set of inputs that instigate state transitions. The state transition function, denoted by δ, is a mapping from the Cartesian product of Hs and $\sum$ to Hs ($Hs \times \sum \rightarrow Hs$). The initial state is represented by s_0.

$$M = (Hs, \sum, \delta, s_0) \tag{4.27}$$

The FSM used can be observed in Figure 4.13, which delineates all managed states, as well as the conditions that govern transitions that are enumerated and explained below.

The ***Lane Keeping*** state is the default operational mode where the vehicle maintains its lane position and cruise control is activated. Transition from this state to others occurs under certain conditions. The transition to the *car following* state is triggered when the vehicle initiates car following procedures, which can occur due to the detection of another vehicle in the same lane (1) or through HMI interaction (2). The *merging* state is entered when the vehicle intends to change lanes, which could be due to the detection of an obstacle (3), HMI interaction (4), an oncoming

roundabout (5), or on-ramp (6) merging scenario. These two last cases, it also need to be detected possible collisions with other vehicles. The *overtaking* state is triggered when the vehicle needs to bypass a slower vehicle or an obstacle but intends to stay in its original lane (7). Lastly, the *parking* state is entered when the vehicle reaches its destination (8).

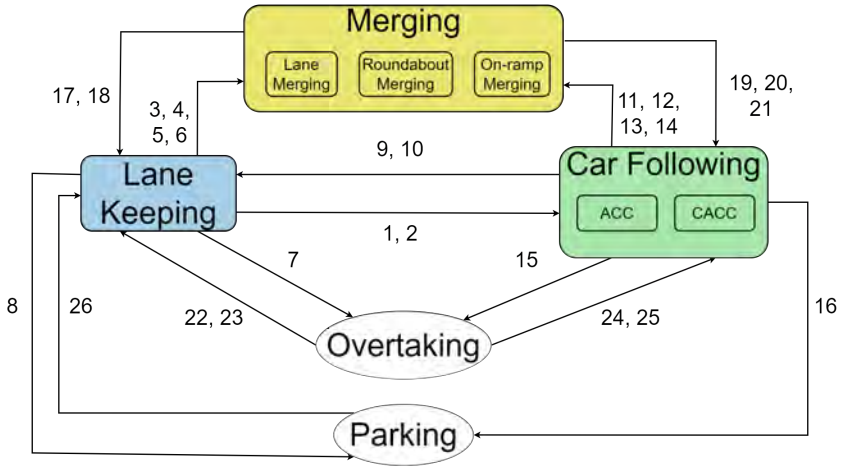


Fig. 4.13: FSM state diagram, where the states are: Lane Keeping, Car Following, Merging, Overtaking, and Parking. The numbers are the conditions to transition from one state to another (explained in the text).

The ***Car Following*** state, which is activated when a vehicle following procedure commences, consists of two sub-states, ACC and CACC. The ACC sub-state is the default state until the vehicle is fully integrated into a platoon or is following a non-cooperative vehicle. The CACC sub-state is activated only when the vehicle is stable within a platoon and is where maneuvers such as platoon merging or overtaking occur. This sub-state reverts to ACC when the communication link is compromised.

A transition from *car following* to *lane keeping* is initiated when the vehicle is no longer part of the platoon while remaining in the same lane (9). Alternatively, a transition may occur if the vehicle ceases to follow the vehicle in front while previously being in ACC mode (10).

The state can also transition to *merging* under certain conditions. If the vehicle is no longer part of the platoon and seeks to change lanes (11). Furthermore, the *car following* state transitions to *merging* when the platoon merging maneuver necessitates the vehicle to reach the main lane. This could include situations such as a lane change (12) or a roundabout entrance (13). Alternatively, the state may transition if the vehicle ceases to follow the preceding vehicle and seeks to alter its current lane while previously being in ACC mode (14).

The *car following* state may also transition to *overtaking* in the event of a required platoon overtaking maneuver (15). This is typically necessary to avoid an obstacle or slower vehicle and the vehicle return to its original lane.

Finally, the *car following* state transitions to *parking* when the vehicle reaches its destination (16), prompting it to leave the platoon and initiate the parking procedure.

The ***Merging*** state is activated when the vehicle changes its current lane and can be executed in cooperative and non-cooperative modes. This state includes sub-states for lane merging, roundabout merging, and on-ramp merging, each corresponding to different road and traffic scenarios.

There are specific circumstances under which the state can transition from *merging* to other states. Firstly, a transition from *merging* to *lane keeping* is initiated when the vehicle has successfully incorporated into the intended lane and is not part of a platoon (17) or when the lane change is aborted (18). Secondly, a transition from *merging* to *car following* is prompted when the vehicle has successfully integrated into the intended lane while being part of a platoon (19), or when it forms a new platoon in the process of incorporation into the intended lane (20). This transition also occurs when a vehicle in a platoon aborts the lane change (21).

The ***Overtaking*** state is activated when the vehicle changes its current lane with the intent to surpass an obstacle or a slower vehicle and return to its original lane. This state differs from the *merging* state as it considers variables such as vehicles in opposite lanes.

The *overtaking* state allows transitions to other states under certain circumstances. A transition from the *overtaking* state to the *lane keeping* state occurs when the overtaking maneuver is completed and the vehicle is not part of a platoon (22), or when the overtaking maneuver is aborted (23). Similarly, a transition from the *overtaking* state to the *car following* state takes place when the overtaking maneuver is completed while the vehicle is part of a platoon (24), or when a vehicle within a platoon aborts the overtaking maneuver (25).

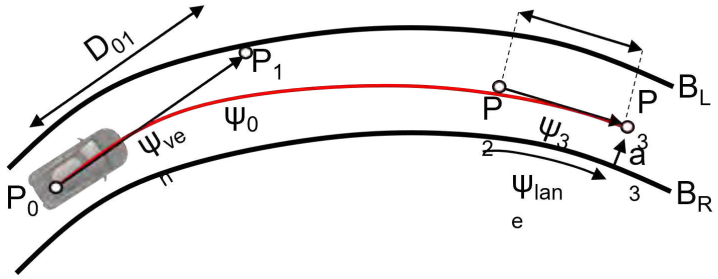
Finally, the ***Parking*** state is activated when the vehicle proceeds to park. Depending on the parking spot, this state can transition to sub-states for Angle Parking, Perpendicular Parking, or Parallel Parking, each representing a different parking configuration. This state transitions to *lane keeping* when the vehicle initiates its driving process (26).

This thesis primarily concentrates on the *lane keeping*, *car following*, and *merging* states, as these are integral to the performance of car following, merging, and platoon merging maneuvers discussed in this chapter.

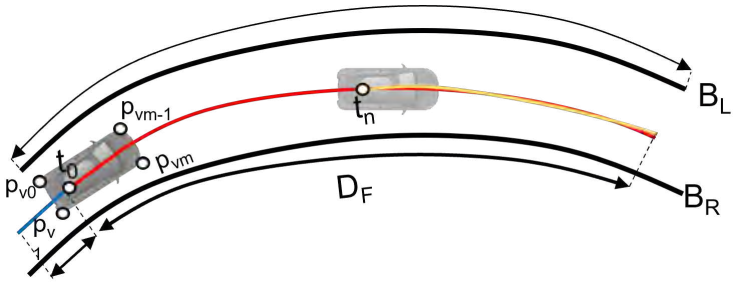
Each state is equipped to process communication messages and perform collision evaluations propagating the position of the vehicles over time with their information, both of which may act as triggers to facilitate transitions between states. Furthermore, each state generates a modified version of the drivable space required by the local planner to produce both the trajectory and the speed profile as detailed in the following section.

4.3.3 RTTP: Local Planner

The method for trajectory generation, derived from the studies of Lattarulo, et al [212, 213], employs the Bound Optimization BY Quadratic Approximation (BOBYQA) method in conjunction with Bézier curves. The method uses certain inputs, such as the lateral bounds B_L and B_R, and total length D, as depicted in Figure 4.14, while taking the following considerations into account.



(a) Bézier Control points example, distance between points and direction of the trajectory.



(b) Trajectory distance considerations and vehicle control points example in $t = 0$ and $t = n$.

Fig. 4.14: Bézier control points positioning example, indicating th directions, the distance between points, as well as the distance considerations in the generated trajectory.

- The bounds are filtered to reduce the density of points, thus ensuring that the computed trajectory only includes significant points. This reduction takes into consideration the gap between consecutive points and the difference in angles.
- The distance D_F is set as the maximum frontal distance, defined by the drivable space, whereas the distance D_R is the rear-end distance employed to form a segment of the line that the vehicle has traversed, thereby facilitating a better fit of the route within the drivable space.
- The first point (P_0) is positioned in the middle of the vehicle's rear axle to minimize the slip angle between the generated trajectory and the vehicle's movement. The direction of the trajectory Ψ_0 at P_0 is identical to that of the vehicle (Ψ_{veh})
- The location of the last point (P_3) is set at a distance D_F from the first point and over the perpendicular lane axis a_3. The position of the control pointP_2 is

established based on the position of P_3, the distance D_{2-3}, and the direction Ψ_3, which is equivalent to the direction of the lane Ψ_{lane} at point P_3.

The optimization process is designed to create an optimal trajectory by varying the following parameters:

- The distance between control point P_0 and P_1, denoted as D_{0-1}.
- The distance between control point P_2 and P_3, denoted as D_{2-3}.
- The distance between the bounds B_L and B_R of control point P_3 along the axis a_3.

The generated lane takes into account the dimensions of the vehicle and ensures that the curvature is reduced below the vehicle's maximum turning limit. This adjustment enhances comfort by mitigating abrupt directional changes in the vehicle. Following the generation of the control points, they are interpolated using Bézier formulation, resulting in a polyline composed of points p_i each associated with a curvature k_i. These points are subsequently evaluated using the cost function as indicated in Equation 4.28.

$$\begin{aligned} \phi &= \begin{cases} \sum_{i=1}^{n} \phi^{-}(p_i, k_i), & \text{when feasible} \\ |\Delta K_0| + \sum_{i=1}^{n} \phi^{+}(p_i, k_i), & \text{when unfeasible} \end{cases} \\ \phi^{-}(p_i, k_i) &= -(\sum j = 1^{m} \min(dR_{pv_j}, dL_{pv_j})) - \frac{1}{|K_i|} \\ \phi^{+}(p_i, k_i) &= (\sum j = 1^{m} \max(dR_{pv_j}, dL_{pv_j})) + |K_{max} - K_i| \end{aligned} \tag{4.28}$$

A trajectory is deemed unfeasible under the following conditions:

- For any of the interpolated points, p_i, if at least one point from the set, pv (points defining the vehicle rectangle), is found outside the lane.
- When the curvature, k_i, at the point, p_i, exceeds the maximum vehicle limit, k_{max}.

Additionally, the discrepancy between the curvature at the starting point and the corresponding point in the previous feasible trajectory, denoted as ΔK_0, must remain below a defined maximum limit.

(dR_{pv_j}, dL_{pv_j}) refer to the distance between the point pv and the right or left bound respectively. The optimal trajectory is determined by evaluating the optimization problem, $\min(\phi)$. A positive value of the cost function signifies an unfeasible trajectory and vice versa. In instances of an unfeasible solution, the optimization seed, as well as the maximum lane distance, are randomly adjusted to aid the convergence of the optimization process.

The BOBYQA method [214] is adopted for the optimization problem resolution, due to its ability to find a reliable local optimum swiftly. Originally a FORTRAN package, BOBYQA is designed to minimize the function $F(x), x \in \Re$, subject to the constraints $a_i \leq x_i \leq b_i : i = 1, 2, ...n$. A key advantage of this method is the

absence of a requirement for pre-computed derivatives, which is beneficial for highly nonlinear problems. The steps involved in solving the optimization problem using BOBYQA are as follows:

- The system checks the boundaries to prevent overlaps or crossings between the lower bound elements (a_i) and upper bound elements (b_i). Furthermore, the initial condition $x(0)$ must lie within these bounds.
- The algorithm generates a quadratic approximation of the objective function $F(x)$ in the form of $Q(x_k) = F(x_k) : k = 1, 2, ..., m$, where m is the number of discretization steps.
- The truncated conjugate gradient method, a variant of the conjugate gradient method with a limited number of iterations, is applied to solve the problem. This method can obtain the optimal value $X*$ in fewer iterations compared to traditional methods [213]..

The solution vector exists in $\Re^3$, where the first element represents the distance between control points P_0 and P_1 of the Bézier curve. The second element signifies the distance between points P_2 and P_3, while the final element pertains to the magnitude of the vector extending from the right to the left boundary, perpendicular to the directionality of point P_3. Notably, the BOBYQA method, as computed through the DLib toolkit, was employed for this process. This toolkit, available in both C++ and Python interfaces, is distributed under a boost open source license.

4.3.4 Speed Planning For Cooperative Maneuvers

The RTTP method is comprehensively equipped with its own speed planner, as delineated in [212]. This method is notable for its ability to generate optimal trajectories across a variety of road components, including but not limited to, roundabouts and lane changes. However, it is not explicitly designed to execute maneuvers that requires the vehicle to adjust its position based on the actions of other vehicles, with the exception of scenarios involving emergency braking or obstacle avoidance. To address this, the RTTP is improve by adding the car following strategy, introduced in Section 4.1. This accommodation allows to modify the speed reference when maneuvers, such as car following itself or merging maneuvers requiring position adjustments, are executed. This transition is facilitated by the behavioral planner when state changes occur. Nevertheless, it is important to note that in the event of an emergency braking maneuver, the speed profile will override any inputs from the Behavioral Planner.

In this scenario, the vehicle's merging sequence adheres to the procedure elaborated in Section 4.2.3. The vehicle planning to merge conducts a virtual platooning with the vehicle in the main lane, adjusting its position over time until it reaches a point where a safe merge is possible. The ego-vehicle and other participants' positions, based on their current speed, acceleration, yaw, etc., are projected onto their current lane using a Kinematic model. This projected data is then used to check

for potential collisions. If no collisions are detected, the vehicle proceeds with the merge. However, if a collision risk is identified, the merge is aborted or the vehicle stops before the merging zone - a strategy specifically applicable in the context of roundabout merging. Being this strategy one of the main contributions, as both decision methods lacks of longitudinal manipulation beside overtaking maneuvers before the start of this thesis.

Furthermore, alongside the generation of trajectories encompassing a speed profile beneficial to cooperative maneuvers, it becomes imperative for vehicles to exchange information beyond mere kinematic data. Consequently, the subsequent section elucidates the maneuver negotiation procedure requisite for achieving platoon maneuvers, offering a comprehensive breakdown of the V2X messages aligned with the maneuvering process elaborated in this thesis.

4.4 Maneuvers Negotiation

Currently, maneuver negotiations on cooperative maneuvers in terms of V2X messaging reflects a dynamic and evolving landscape, driven by international efforts to establish common standards and protocols that facilitate effective communication between vehicles and their environment. There is growing interest in defining specific messages for coordinating maneuvers in complex traffic situations, such as intersections, car following and mergers, with the aim of improving road safety and traffic efficiency. Additionally, discussions also focus on V2I communication, such as smart traffic lights, seeking to enhance coordination to optimize traffic flow.

However, given the absence of a clear directive regarding the communication protocol for executing cooperative maneuvers, this study derives its approach from the standard defined by the ENSEMBLE project [215]. The defined messages, designed for platooning operations, adhere to the ETSI format and fall into two categories: Platoon Management Messages (PMM) and Platoon Control Messages (PCM). While these messages are intended to be in the facilities layer, alongside messages such as CAM or DENM, they are encapsulated into a GN PDU format, positioning them in the Networking and Transport Layer. This encapsulation ensures that these messages use the header and security measures outlined in the ETSI EN 302 636-4-1 V1.3.1 [216] document for GN messages. More specifically, GN messages are chosen to facilitate the exchange of information with specific vehicles of the platoons, thereby minimizing unnecessary message distribution among other vehicles.

For these messages to be used, the vehicle must first receive CAM messages from surrounding vehicles. Upon receipt, the vehicle searches for other vehicles in front within a predetermined distance. If a MAC address is obtained and the vehicle is deemed of interest, the negotiation process is initiated.

PMM play a crucial role in establishing the negotiation process between vehicles and/or platoons. These messages encapsulate information pertaining to join/leave or request/response, which are disseminated at a frequency of 10 hz when needed. The content of these messages is elaborated in Table 4.4 for further understanding.

Tab. 4.4: PMM messages description.

Container	Content		Description	Data Type	Unit
header	Its Pdu Header		Header structure for GN messages.	-	-
reference Position	position		Longitude, latitude and altitude in WGS84 coordinates.	int32_t	$0.1\mu°$
	heading		Heading in WGS84 coordinate systems.	uint16_t	0.1°
message Choice	join Request		Request to join the platoon	bool	-
	join Response	status	Response to joining request. - 0: Not allowed. - 1: Accepted, but the vehicle needs re-position. - 2: Fully accepted.	int	-
		platoon ID	The ID of the platoon.	uint32_t	-
		joining At Position	Position where the vehicle is joining, from 1 to n	int	-
	joining Process		Joining process of the vehicle: - 0: Adjusting from behind. - 1: Adjusting from an adjacent lane. - 2: Adjusting process finished.	int	-
	leave Request	status	Status of the leaving process. - 0: Not leaving. - 1: Started. - 2: Finished.	int	-
		platoon Position	A vehicle position in a platoon, from 1 to n	int	-

Upon the successful completion of negotiation and formation of a platoon, the **PCM** are dispatched. These messages are devised to exchange comprehensive information about the vehicle's dynamics and kinematics. The depth of these messages lies in the variables of the vehicles, which incorporate data such as reference

speed or reference acceleration, thereby enhancing the performance of the platoon. These messages are disseminated at a frequency of 20 hz during the platoon driving. An extensive description of the content of these messages can be found in 4.5.

Tab. 4.5: PCM messages description.

Container	**Content**		**Description**	**Data Type**	**Unit**
header	Its Pdu Header		Header structure for GN messages.	-	-
reference Position	position		Longitude, latitude, and altitude in WGS84 coordinates.	int32_t	$0.1\mu°$
	heading		Heading in WGS84 coordinates.	uint16_t	0.1°
paltoon Control	vehicle In Front ID		Predecessor vehicle ID.	int	-
	longitudinal Control Container	longitudinal Acceleration	Long. acc.	int	$0.1\frac{m}{s^2}$
		speed	Current vehicle speed.	uint16_t	$0.01\frac{m}{s}$
		reference Speed	Reference speed.	uint16_t	$0.01\frac{m}{s}$
		predicted Longitudinal Acceleration	Reference longitudinal acceleration.	int16_t	$0.1\frac{m}{s^2}$
	lateral Control Container	lateral Acceleration	Lateral acceleration.	int16_t	$0.1\frac{m}{s^2}$
		steering Wheel Angle	steering wheel angle.	uint16_t	1.5°
		curvature	Vehicle position curvature.	int32_t	$\frac{1}{30000}m$
		yaw Rate	Vehicle yaw rate.	uint16_t	$0.01\frac{°}{s}$
	open Gap		Gap opening order.	bool	-

To elucidate the functionality of the aforementioned messages, Figure 4.15 shows the flow state diagram where vehicles situated in the same lane aim to form a platoon. This process begins when a vehicle detects no vehicle ahead and receives a join request through the PMM. In response, this vehicle assesses the road conditions. If deemed acceptable, the vehicle acknowledges the request by transmitting a corresponding PMM, thereby assuming the role of the platoon leader. For the follower vehicle,

once the join request is approved, it initiates the car following strategy delineated in Section 4.1, where enters in ACC mode to reduce the gap until the reference distance is achieved. Subsequently, the follower vehicle communicates to the leader that the joining process has been completed and begins transmitting PCM. Thereafter, the vehicle transitions to the CACC mode, enabling platoon driving. In the event that the join request is declined or receives no response, the vehicle will wait a predetermined duration before re-sending the request. This process is repeated until the maximum limit is reached, at which point the vehicle aborts the attempt to form the platoon and reverts to the *lane keeping* state. The platoon is disbanded when the vehicles reach their respective destinations or when a leave request is transmitted via a PMM. In the latter scenario, if the vehicle is located in the middle positions, the gap remains open until the leaving procedure is completed, a process that is also communicated via a PMM. Subsequently, the gap is closed. If the vehicle is in the last position, it exits without any required maneuver, similar to a vehicle in the first position. However, in the case of the first vehicle, the second vehicle subsequently assumes the role of the platoon leader.

In the event that a vehicle or a platoon is situated in an adjacent lane, the same communication messages are used, although the sequence differs, as demonstrated in Figure 4.16. Specifically, when an obstruction in the route is identified, for instance, the platoon in that lane (termed as Platoon B) issues a request to join the neighboring platoon (designated as Platoon A) via a PMM. If this request is approved, Platoon B embarks on a position adjustment, aligning with the space previously vacated by Platoon A. In order to facilitate this space, the leader of Platoon A identifies the locations of the vehicles in Platoon B. If these vehicles are in close vicinity, the leader dispatches a PCM to Platoon A, initiating an open gap action. Once the vehicles in Platoon B have successfully adjusted their positions and completed the lane change, a new platoon, referred to as Platoon C, is formed. Here, the leader is the foremost vehicle, and the platoon position of each vehicle is re-designated accordingly. In contrast, if the request is declined or receives no response, the leader of Platoon B continues to issue additional requests until the limit is reached, subsequently resulting in a stoppage at the conflict point.

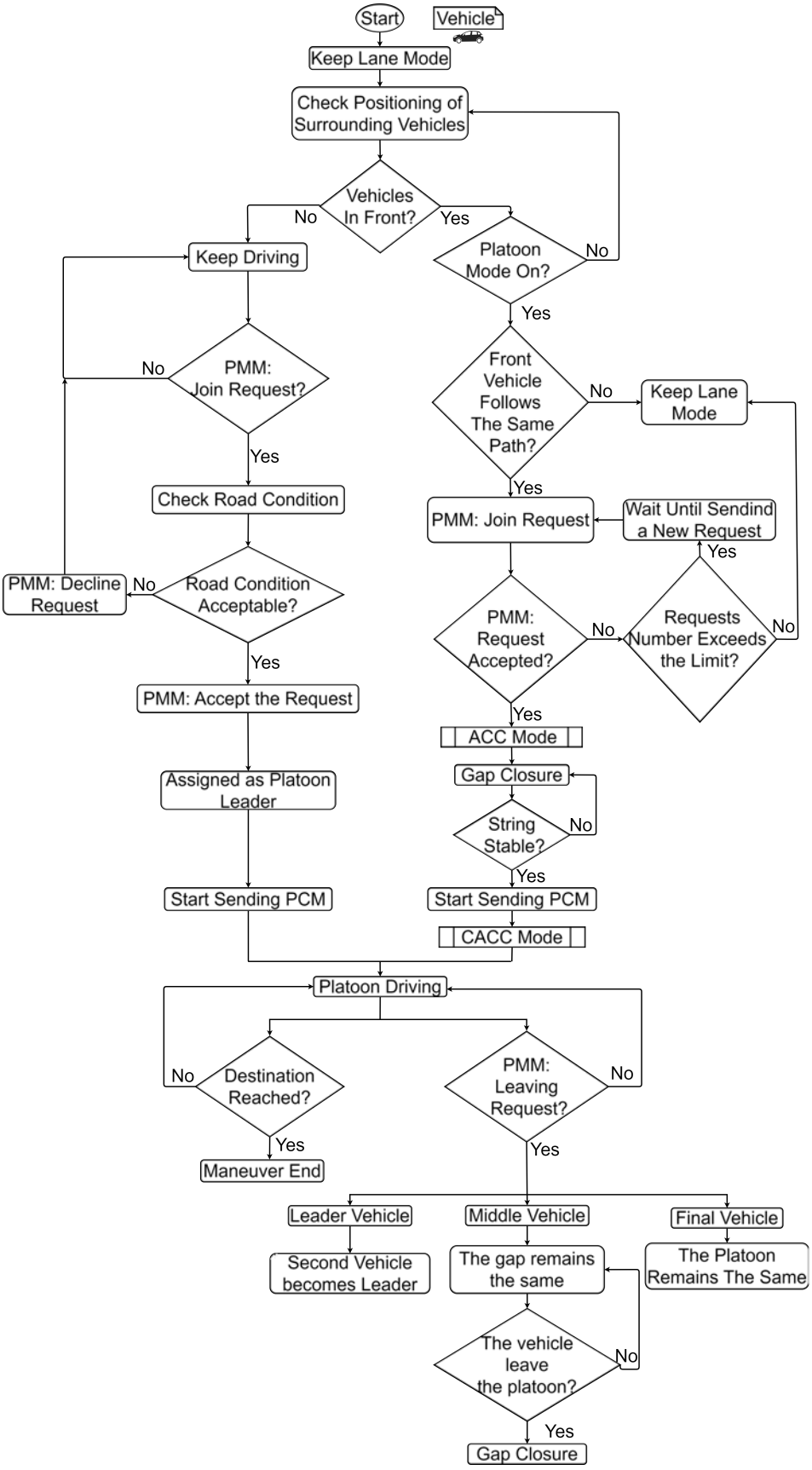


Fig. 4.15: Flow state diagram of the platoon driving, from the initial negotiation point until the end of the maneuver. This case is presented when the vehicles are located in the same lane.

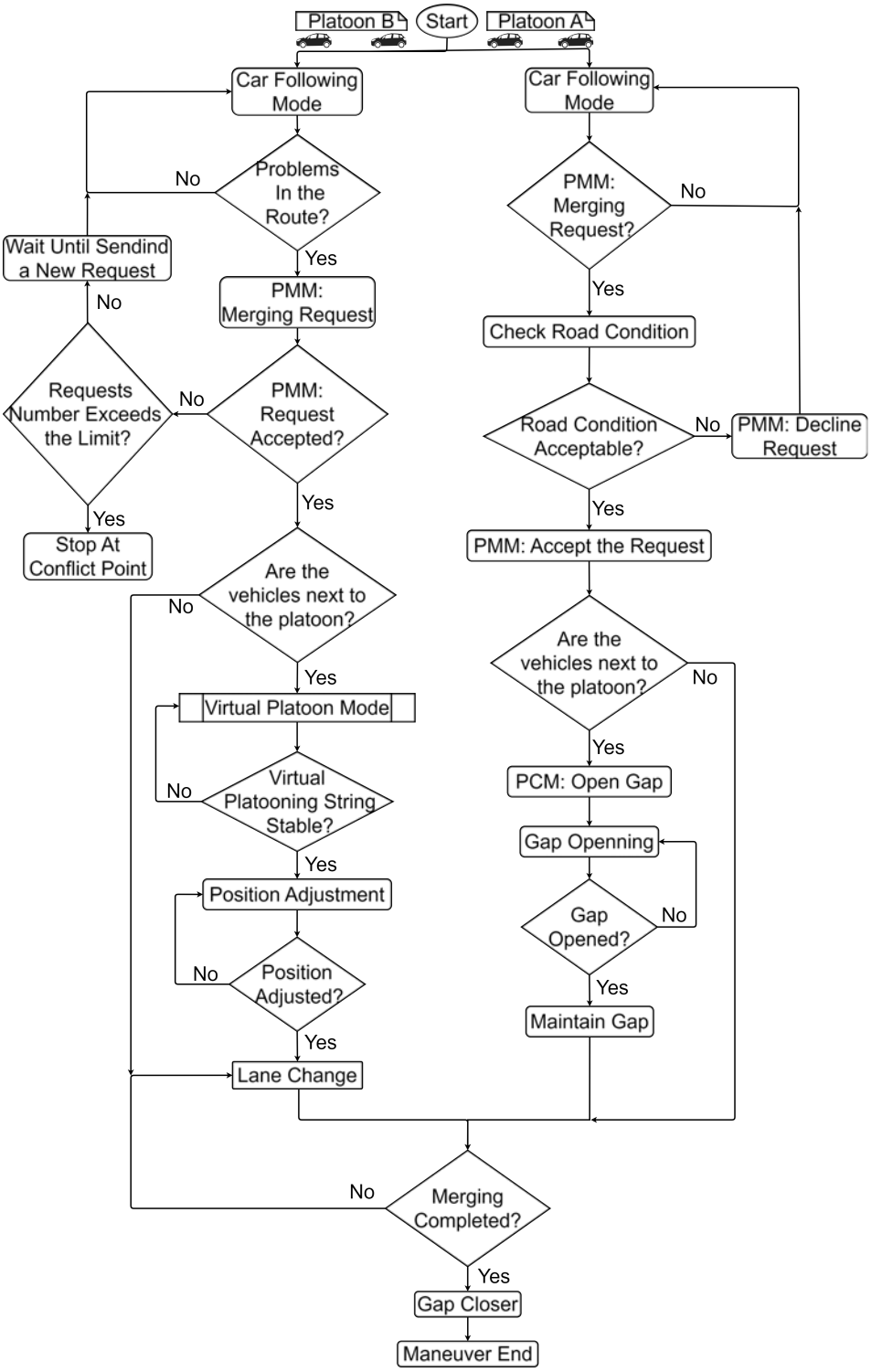


Fig. 4.16: Flow state diagram of the platoon merging, from the initial negotiation point until the end of the maneuver. This case is presented when the vehicles are located in different lanes.

4.5 Summary and Conclusions

This chapter introduces two decision methodologies designed for the execution of cooperative maneuvers. The first strategy, HYTP, incorporates nominal static trajectories based on 4th and 5th order Bézier Curves, adjusted by a predictive maneuver planner. This planner has been adapted in this thesis to execute maneuvers such as overtaking and merging. The second approach, RTTP, employs a detailed map and an A* search algorithm to generate a drivable space. This is then modified by a behavioral planner, which includes a FSM with various states representing maneuvers, notably the *lane keeping*, *car following*, and *merging states*. A local planner that uses real-time trajectories based on 3rd order Bézier curves is also incorporated. The latter method allows for the execution of more complex scenarios and possesses a more defined structure, aligning more closely with navigational processes outlined in existing literature.

Following the description of both decision methodologies, the logic guiding the negotiation of maneuvers is presented, introducing the V2X messages employed, namely the PMM and PCM. These messages are based on compressed versions of those used in the ENSEMBLE project. This chapter also presents the car following strategy, which is used in both decision methods and incorporates the use of ACC and CACC technologies, both of which involve a feedforward/feedback mechanism with PD control. The Renault Twizy 80 simulation model in the Dynacar simulator is used for the description of this strategy.

The HYTP method has been implemented in two publications [170, 217], where roundabout were tested. The second method led to one publication [218], where an initial version of the FSM strategy was employed to respond to obstacles shared by cloud services.

In the succeeding chapter, the standalone car collowing strategy, as well as both decision methods for merging maneuvers, will be evaluated. This evaluation will offer a performance comparison of both strategies. Furthermore, the maneuvers used to test the SerIoT and IoTAC systems will be evaluated, also showcasing the performance of both cyber-security systems.

5 Results and Discussions

"I wanna be the very best Like no one ever was To catch them is my real test To train them is my cause." Gotta catch 'em all! - Jason Paige

Vehicle cooperation has been identified as a promising approach to manage the complex task of driving in intricate scenarios. For it to emerge as a successful solution, it is crucial to validate it across diverse environments, particularly those that replicate real-world conditions. This chapter presents the outcomes of validation tests for the approaches delineated in Chapter 4 for each maneuver, using the framework, platforms, and scenarios described in Section 3.

To provide a comprehensive understanding of the progression and motivation of the results achieved in this thesis, Figure 5.1 illustrates a chronological sequence of the most significant milestones. These milestones are contextualized within various national and international research projects in which they were accomplished.

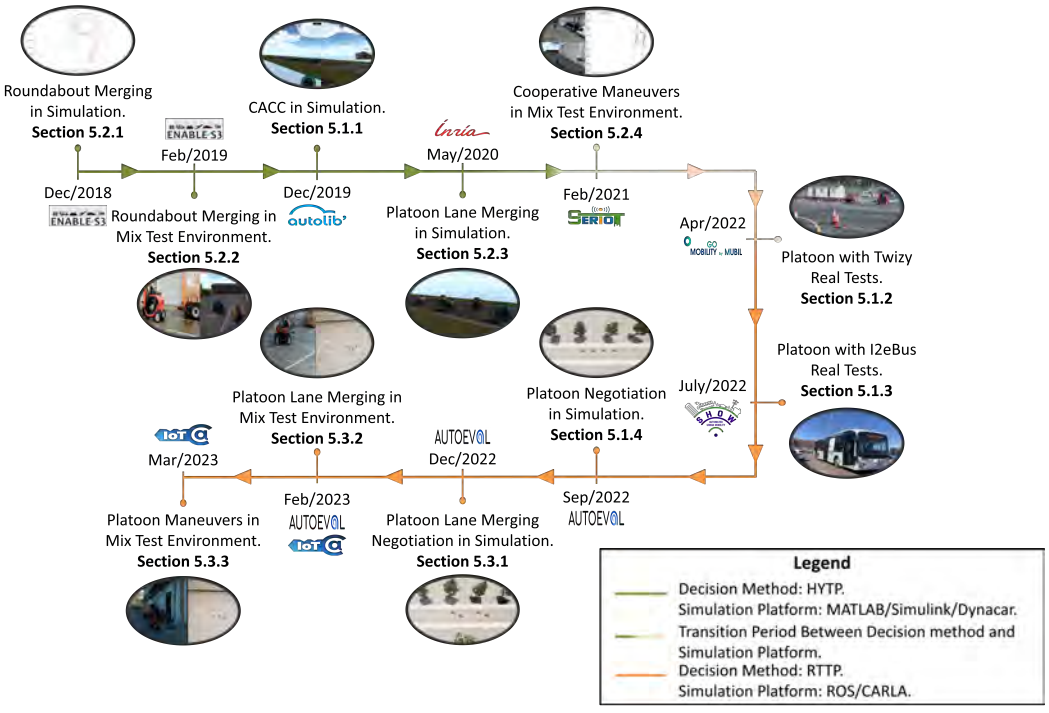


Fig. 5.1: Time-line of each result accomplishment in concordance with the respective project.

The timeline shows that between 2018 and 2021, efforts were primarily concentrated on the development of the car following controller described in Section 4.1,

and the exploration of the HYTP approach outlined in Section 4.2 for merging maneuvers, using the MATLAB/Simulink + Dynacar simulation environment of Section 3.2.1.1. The period from 2021 to 2022 marked a transition phase due to the limitations identified in both the decision method and the simulation environment, leading to the activities in the second half of the timeline. From 2022 to 2023, work focused on the implementation of the RTTP method presented in Section 4.3 and the integration of the car following strategy using the ROS + Carla environment described in Section 3.2.1.2.

Further sections provide detailed insights into each result and their motivating factors, starting with results pertaining to the car following strategy, which incorporate tests in both simulation environments as well as in a Renault Twizy and the Irizar i2eBus.

The subsequent section present the results of various maneuvers using the HYTP approach, demonstrating roundabout merging in both simulation and mixed scenarios, platoon lane merging in a simulated environment, and fleet management and intersection management in a mixed scenario to validate the cyber-security framework of the SerIoT project.

Finally, the outcomes of the RTTP are presented, showcasing a simulated platoon lane merging and a mixed environment test of the same maneuver. Building upon the IoTAC project framework, multiple iterations of the aforementioned merging maneuver were conducted, with a distinct focus on a case where the actual vehicle integrates from the rear rather than the middle. These tests examine the cyber-security capabilities of the IoTAC system in the field of CCAM.

5.1 Car Following Strategy

The first phase of assessing the effectiveness of the car following strategy for car following maneuvers, as outlined in Section 4.1, including the negotiation process explained in Section 4.4. It involves a comprehensive review of the control strategy detailed in the same section. Throughout the course of this research, four significant milestones were accomplished to fulfill this objective. Each of these milestones is visually represented in Figure 5.2. The following enumeration provides further details on each validating test.

1. The car following controller for both, ACC and CACC technologies have been implemented in a simulated straight line. This implementation, lays a groundwork for tuning these maneuvers in other environments and platforms (refer to Figure 5.2a).
2. The car following controller has been implemented with real Twizy vehicles in a closed circuit. This evidences the successful application of the controller in real-world environments (refer to Figure 5.2b).

3. The car following controller has been implemented with the i2eBus, demonstrating the adaptability of the approach as it was applied to different platforms (refer to Figure 5.2c).
4. A platoon has been formed according to the complete car following strategy in a simulation. This marks the successful execution of the decision, negotiation and control process for forming a platoon of vehicles in complex urban environments (refer to Figure 5.2d).

(a) 4 vehicles platoon in the Dyancar Simulator in a straight line scenario.

(b) 2 Twizy platoon in the Ficoba' test track under the 2022 Go Mobility event.

(c) Irizar I2eBus and Gulliver shuttle platoon in the EMT Carabanchel depot.

(d) 4 vehicles platoon in CARLA Simulator in the Tecnalia's vicinity scenario.

Fig. 5.2: Images of each test performed to validate the car following strategy.

It is important to emphasize that each implementation required a tuning process to determine the most suitable parameters for each situation. Following the methodology outlined in Section 4.1, the derived parameters of each platform are presented in Table 5.1.

The following sections will present the results corresponding to each test, accompanied by detailed descriptions of the methodologies employed in their execution.

5.1.1 ACC and CACC Simulated Performance

Four Twizy vehicles are simulated to assess the performance of both ACC and CACC technologies in a linear scenario, with no negotiation process involved. The primary objective of these tests is to evaluate controller performance within the longitudinal domain and establish a tuning methodology for the controllers. To

Tab. 5.1: Car following configuration parameters for each platform.

	Platform			
Parameter	MATLAB/Dynacar	Twizy	i2eBus	ROS/CARLA
K_p	0.5393	0.8393	2.2	0.5
K_d	0.4103	0.4103	0.4	0.4
d_{std}	3 m	3 m	7 m	3 m
h_{min} - h_{max}	0.6 - 1.2 s	0.6 - 1.2 s	0.8 - 1.2 s	0.6 - 1.2 s

ensure accurate representation of the Twizy, these tests are conducted using the Dynacar simulator, which offers an accurate vehicle model [89]. This approach alleviates the extrapolation of the algorithms into other scenarios.

The follower vehicles are first required to attain the minimum standstill distance of 3 m, after which the leading vehicle start driving. The initial vehicle adhered to a speed profile with numerous speed alterations, peaking at 8.33 m/s to observe the evolution of the inter-distance of the follower vehicles and the time gap outputted by the algorithm for both scenarios. It is important to note that communication delays are assumed to be zero seconds in these tests.

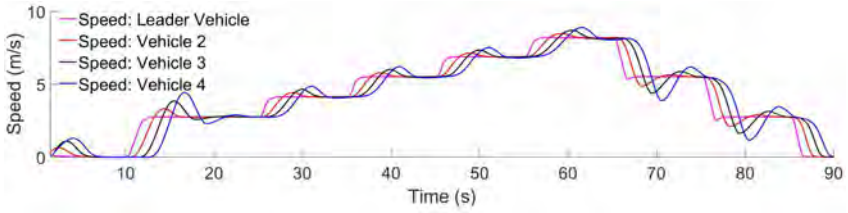
Figure 5.3 illustrates the performance of the ACC, presenting speed (Figure 5.3a), distance (Figure 5.3b), and time gap (Figure 5.3c) tracking. The ACC's performance, establishing in the references, propagated the speed changes of the leading vehicle with the most sudden changes (2.77 m/s) presenting the most errors, peaking at 1.61 m/s, 2 m and 0.45 s for the last vehicle. However, the strategy's performance is adequate for car following maneuvers when there is no communication availability or while the engagement procedure is being conducted.

Figure 5.4 displays the performance of the CACC, presenting the speed (Figure 5.4a), distance (Figure 5.4b), and time gap (Figure 5.4c) tracking performance. In this scenario, as anticipated, the errors are not propagated. The second vehicle in the platoon showed slightly more error, as seen in the time gap tracking, where errors up to 0.31 s are observed. This phenomenon is a result of the second vehicle being the most reactive of the followers, as the 3rd and 4th vehicles utilize the references of the preceding vehicle, which is being progressively attenuated with each vehicle.

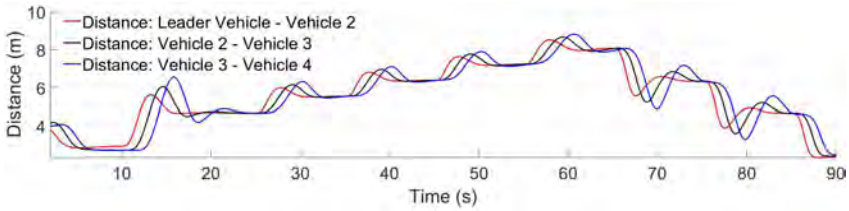
5.1.2 Renault Twizy 80 Real Test Performance

Upon successful validation of the car following controller within simulated environments for ACC and CACC technologies, it is subsequently applied to real-world scenarios using two Twizy vehicles, which also served as the basis for the simulation (refer to Section 3.2.2.1). The lead vehicle, manually operated, is equipped with the architecture outlined in Section 3.1, enabling it to transmit information via V2X communication. Conversely, the following vehicle, assigned to execute the

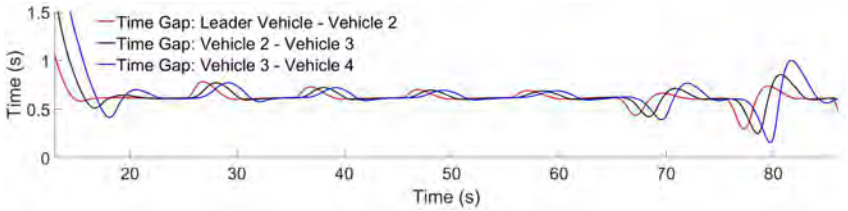
car following maneuver, is in automated mode, utilizing the same architectural framework.



(a) ACC speed tracking performance of the simulated platoon with 4 vehicles.

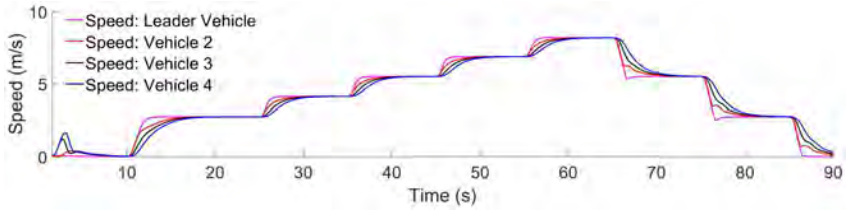


(b) ACC distance tracking performance of the simulated platoon with 4 vehicles.

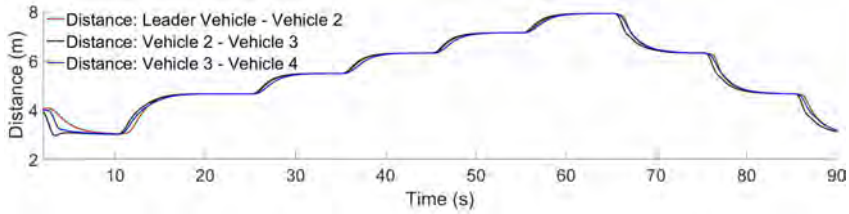


(c) ACC time gap tracking performance of the simulated platoon with 4 vehicles.

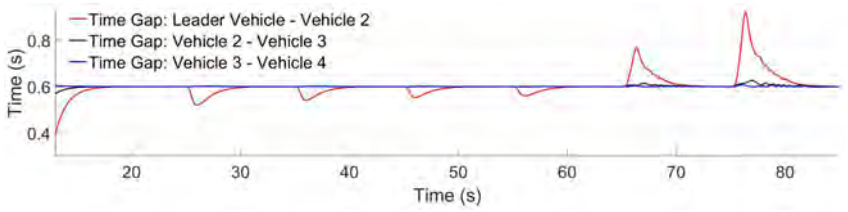
Fig. 5.3: Simulated performance of a platoon of 4 vehicles using ACC.



(a) CACC speed tracking performance of the simulated platoon with 4 vehicles.



(b) CACC distance tracking performance of the simulated platoon with 4 vehicles.

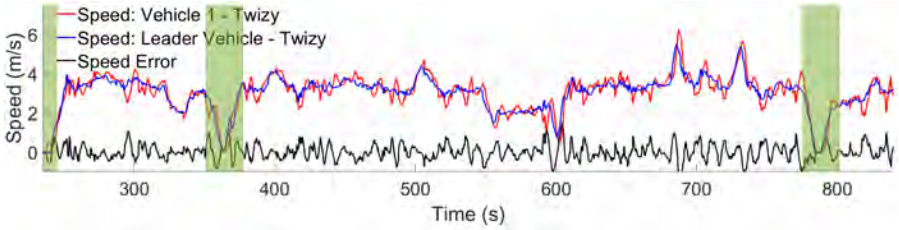


(c) CACC time gap tracking performance of the simulated platoon with 4 vehicles.

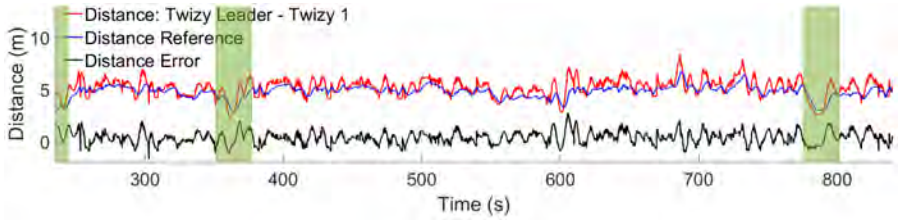
Fig. 5.4: Simulated performance of a 4 vehicle platoon using CACC.

These tests were conducted on the Ficoba test track circuit during the 2022 Go Mobility Fest, providing a public showcase of car following maneuvers. The track's speed limit was set at $20km/h$, which allowed the vehicles to demonstrate a more realistic driving style, incorporating various stops. This added a layer of complexity to the results obtained.

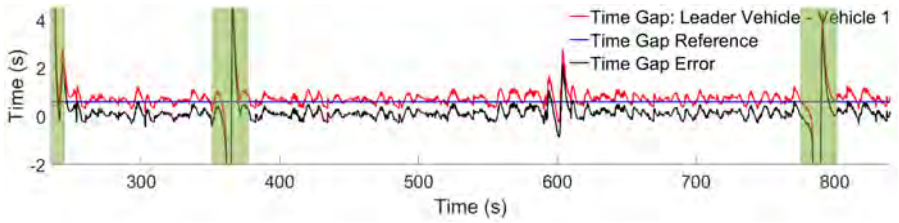
Figure 5.5 shows the performance of the maneuver. Metrics evaluated in these tests are speed (Figure 5.5a), distance (Figure 5.5b), and time gap (Figure 5.5c) between vehicles. The results confirm the effective functioning of the algorithm, as the follower vehicle successfully maintained the leader's speed while closely adhering to the reference distance and time gap. The peaks observed in Figure 5.5c, deriving from the vehicles stopping for demonstration purposes (highlighted with green rectangles in each Figure), led the calculation of the time gap to infinity. However, this situation don't adversely affect the maneuver's performance. Notably, the largest peaks in terms of distance between vehicles occurred during the execution of tight curves on the test track. The maximum error recorded is 2.2 m, which is promptly rectified and do not pose significant issues during testing.



(a) Platoon speed tracking performance with two Renault Twizy.



(b) Platoon distance tracking performance with two Renault Twizy.



(c) Platoon time gap tracking performance with two Renault Twizy.

Fig. 5.5: Real car following performance with a platoon of two Renault Twizy.

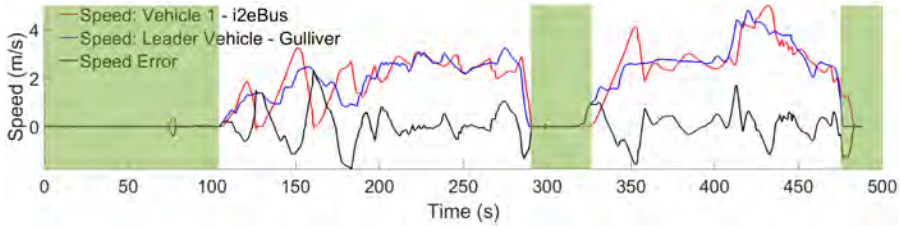
5.1.3 Irizar i2eBus Real Test Performance

Following the effective implementation of the strategy with the Twizy vehicles, the goal was then expanded to include its application to a significantly different platform, thereby testing the adaptability of the strategy as part of the modularity

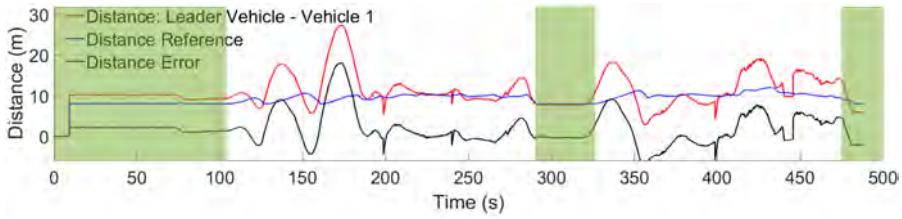
of the AUDRIC architecture. This move was in line with the UC18 - Platoon - of the Madrid Mega pilot within the SHOW project, which investigated an optimization strategy for bus operations within depots, incorporating car following maneuvers. This involved a leader bus, operated manually, directing each follower bus to their designated parking areas.

In this case, the platform is a 12-m Irizar i2eBus (refer to Section 3.2.2.2), which is programmed to follow a Gulliver shuttle in the Carabanchel EMT depot (refer to Section 3.2.4).

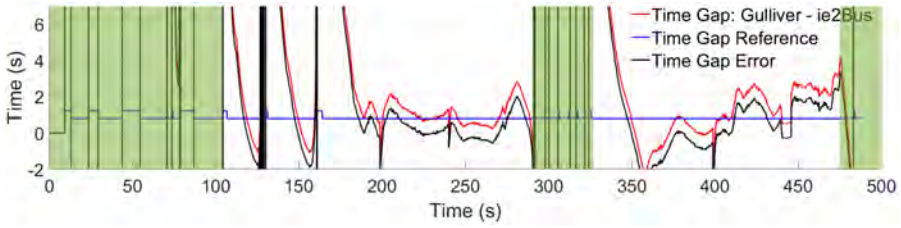
Figure 5.6 exhibits the performance of the car following maneuver. The evaluation metrics employed include speed (Figure 5.6a), distance (Figure 5.6b), and time gap (Figure 5.6c) between the vehicles. The figures indicate the suitable operation of the algorithm, demonstrated by the bus's ability to maintain the leading vehicle's speed and concurrently adhere to the reference distance and time gap.



(a) Platoon speed tracking performance with the Irizar I2eBus .



(b) Platoon distance tracking performance with the Irizar I2eBus.



(c) Platoon time gap tracking performance with the Irizar I2eBus.

Fig. 5.6: Real car following performance with a platoon of one Irizar I2eBus and one Gulliver.

The peaks observed in Figure 5.6c, similar to those in previous tests, are a consequence of the vehicles stopping as part of the testing procedure (zones highlighted with a green rectangle in the figures) or due to a particular case of the vehicle stabilizing, which leads to the calculation of the time gap extending towards infinity. Despite this, the overall performance of the maneuver is not negatively impacted. It should be noted that larger peaks of errors were observed during these tests, predominantly when the bus initiated from a stationary state. Overcoming the bus's

inertia with its current automation proved to be challenging, however, it did not pose a risk as the bus was a minimum of 4 m away from the Gulliver. Even in the most extreme cases, the bus was able to stabilize within 80 s seconds.

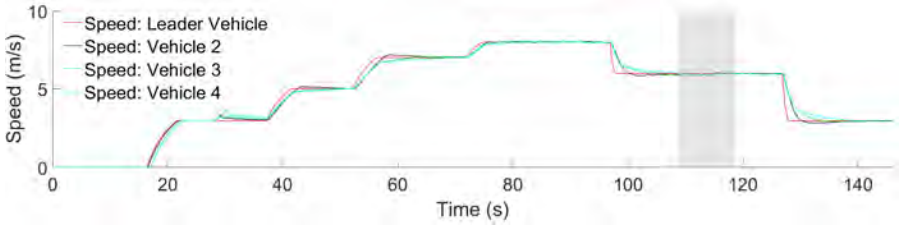
5.1.4 Car Following Strategy Simulated Performance

The culmination of validating the car following strategy, subsequent to evaluating the controller's performance in both simulated and real platforms, is the assessment of the platoon's functionality within an urban context comprising diverse road segments and the involvement of a negotiation process. These tests were executed as a component of UC 2.1 - Advance Platoon Operations - within the AUTOEV@l project, an initiative examining platooning applications in urban environment.

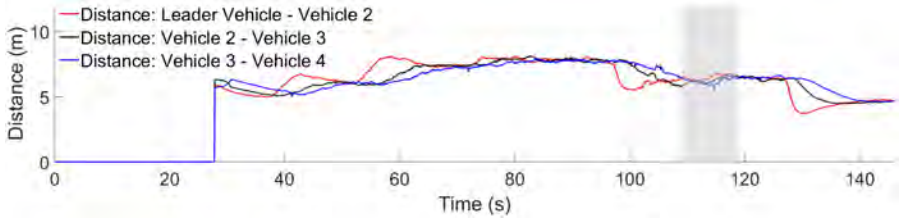
The car following control algorithm is integrated with the RTTP method. This integration is particularly crucial for transitioning between maneuvers.

For this evaluation, the CARLA simulator was used, simulating four vehicles modeled after the Twizy, with a simulated delay of 100 ms. The test starts with the vehicles operating in automated mode. Upon activation of the platoon engagement command, the vehicles formed a platoon. Initially, the vehicles operated in ACC mode. Once they reached the reference distance and time gap, they switched to CACC mode and continued towards their destination.

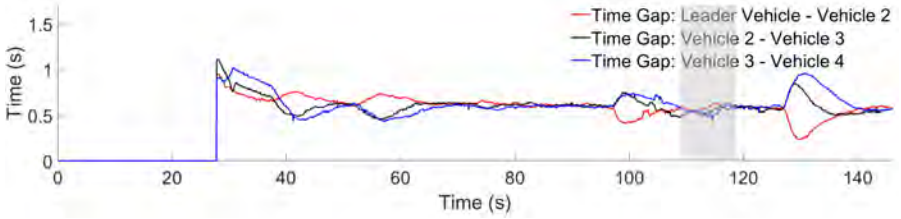
Figure 5.7 demonstrates the performance of the car following maneuver, specifically evaluating parameters such as speed (Figure5.7a), inter-vehicle distance (Figure 5.7b), time gap (Figure 5.7c), and the states of the FSM for each vehicle (Figure 5.7d). The results indicate that at the 16 s, the vehicles start automated driving in the *lane keeping* state. At the second 27, the vehicles initiate the platoon engagement, transitioning to the *car following* state. Within this state, the vehicles initially operate under ACC until they reach the time gap reference of 0.6 s, at which point they switch to the CACC, thus completing the platoon engagement. Subsequently, varying speed changes introduced in the leader vehicle are observed. Notably, the algorithm performs consistently, without propagating speed changes, irrespective of the presence of roundabouts or other curved road segments. This is exemplified by the behavior observed between seconds 109 and 117 (highlighted with a gray rectangle in the figures), corresponding to the vehicle's entry into and exit from the roundabout, where no significant speed changes in platoon are detected.



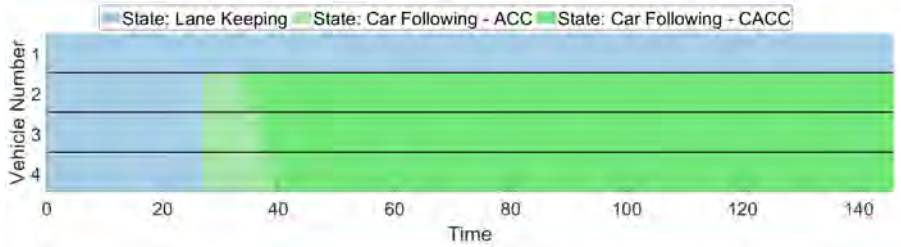
(a) car following speed tracking performance of the simulated platoon with 4 vehicles.



(b) car following distance tracking performance of the simulated platoon with 4 vehicles.



(c) car following time gap tracking performance of the simulated platoon with 4 vehicles.



(d) FSM state transition of each vehicle.

Fig. 5.7: Simulated performance of a 4 vehicle platoon using the car following strategy in an urban environment with different road components.

5.1.5 Discussions

A series of tests, covering both virtual and real platforms, have been undertaken to validate the car following strategy proposed in this thesis. Table 5.2 presents the median error values of each evaluated metric corresponding to each test, namely the speed error between the vehicle and the preceding vehicle, the distance error between the vehicles and the reference distance, and finally the time gap error between the actual time gap of the vehicle and the reference one.

An analysis of the table reveals that the car following controller, specifically in CACC mode, consistently displays low errors across all tests. The lowest error rates were observed in tests performed within the MATLAB/Simulink + Dyancar environment. This can be attributed to the less complex scenario, absence of

Tab. 5.2: Car following mean errors of each test.

Platform	Vehicle	Speed Error (m/s)	Distance Error (m)	Time Gap Error (s)
Simulation by MATLAB/SIMULINK + DYANCAR	Vehicle 2	0.1456	0.18	0.0159
	Vehicle 3	0.1379	0.08	0.0135
	Vehicle 4	0.13	0.025	0.0127
Real Platform	Twizy	0.22	0.42	0.1975
	i2eBus	0.2921	2.3	1.07
Simulation by ROS + CARLA	Vehicle 2	0.1462	0.1722	0.0376
	Vehicle 3	0.1276	0.1482	0.0316
	Vehicle 4	0.117	0.1296	0.0275

curve segments, and the usage of a highly accurate vehicle model. These beneficial circumstances contribute to the efficient implementation and fine-tuning of the controller, which was the objective of these first tests.

The performance of the Twizy in real tests demonstrated the feasibility of implementing this controller in real-world conditions. This was evidenced by the low error rates across all metrics, particularly an error of less than 0.5 m in the vehicle's inter-distance. In contrast, the i2eBus tests demonstrated greater error in the distance among vehicles and the time gap, indicating difficulties in reaching both references, despite the low speed error value of 0.2 m/s. These results could potentially be improved in the future through the acquisition of a more accurate bus model, enabling a better fit through the tuning methodology.

Contrary to being discouraging, these results illustrate the remarkable adaptability of the controller and driving architecture to a variety of platforms. Furthermore, considering the lack of research on the extrapolation of car following maneuvers to vehicles other than trucks or small ones, these findings provide a novel opportunity to explore other types of ground vehicles.

Finally, the simulation of the car following strategy in a more complex urban environment also yielded promising results, as evidenced by the low values across all metrics. While this simulation demonstrated slightly higher error rates than the MATLAB/Simulink + Dynacar environment, it incorporated a negotiation process while driving in curve segments and a different virtual model of the Twizy. Furthermore, it showcased the strengths of the FSM in managing the decision aspects of the vehicle.

Having corroborated the strengths of the car following strategy, the forthcoming sections will delve deeper into the results using the two decision methods for other cooperative maneuvers, which also incorporate the use of this car following strategy.

5.2 Hybrid Trajectory Planning

At the beginning of this thesis, the AUDRIC architecture's primary decision method was based on the HYTP, which was predominantly used for lane change and overtaking maneuvers. To increase the versatility of this approach, a broader spectrum of maneuvers were explored, with a particular emphasis on those related to merging scenarios due to their complex nature. This led to four significant milestones. The detailed enumeration below elucidates each validating test.

1. The application of HYTP for roundabout merging in simulated scenarios demonstrated the feasibility of using this method for maneuvers beyond lane change and overtaking.
2. The execution of HYTP for roundabout merging in mixed test environments provided evidence of the strategy's applicability in real-world conditions.
3. The combination of the HYTP decision method with the car following controller for platoon lane merging maneuvers in simulations represents the successful integration of the two main developments from the first half of the thesis. It also provides a merging strategy that can be extrapolated to the RTTP approach.
4. The implementation of fleet management and smart intersection maneuvers, using the HYTP approach in mixed test environments, demonstrate the capabilities of the SerIoT system in CCAM applications. This also offers an opportunity to explore how infrastructure can enhance cooperative maneuvers.

The following sections will present the results corresponding to each test, accompanied by detailed descriptions of the methodologies employed in their execution.

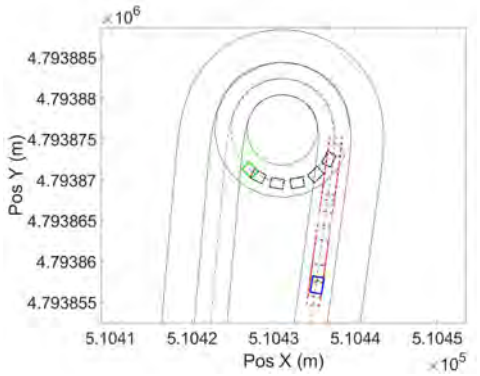
5.2.1 Roundabout Merging Simulated Performance

The initial maneuver assessed within the HYTP strategy is the roundabout merging, as part of the ENABLE S3 project, where methodologies for validating and verifying where studied for different transport domains.

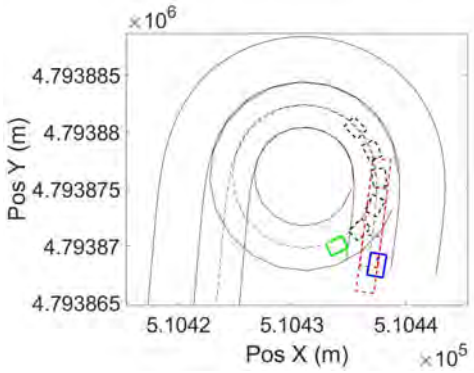
This test focused on a specific scenario where a manually-driven vehicle, equipped with the ability to share its information, is already within the roundabout. Meanwhile, a CAV attempts to enter the roundabout, yielding at the entrance to allow the manually-driven vehicle to pass before joining the roundabout itself.

The initial testing of this maneuver was conducted in a simulated environment as part of the validation methodology employed in the project, utilizing the Dynacar simulator to replicate a single-lane roundabout scenario with two Renault Twizys. An image sequence of the merging process, as depicted in Figure 5.8a, demonstrates the projections of the vehicles (with red rectangles representing the ego vehicle and black rectangles denoting the vehicle inside). This sequence led to a slowdown of the ego vehicle until it applied brakes (Figure 5.8b), yielding space to the other

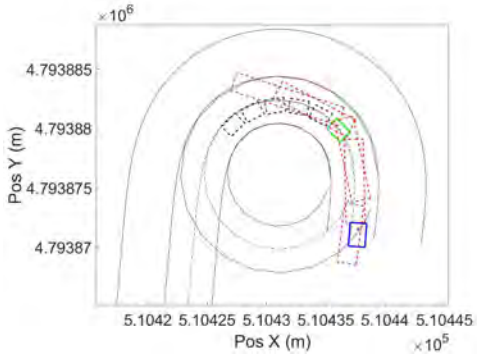
vehicle in the roundabout (Figure 5.8c). Upon ensuring safety, the vehicle resumed acceleration (Figure 5.8d).



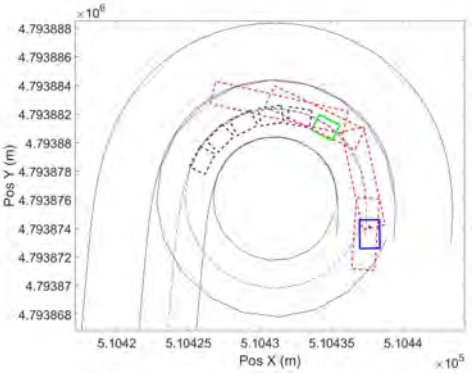
(a) Both Vehicle projections coinciding at the entrance ($t = 11\ s$).



(b) Ego vehicle stops at the entrance of the roundabout to let the inside vehicle pass ($t = 13\ s$).



(c) Vehicle accelerating ($t = 15\ s$).



(d) Vehicle at the roundabout ($t = 17\ s$).

Fig. 5.8: Simulated roundabout merging using the HYTP approach with two vehicles.

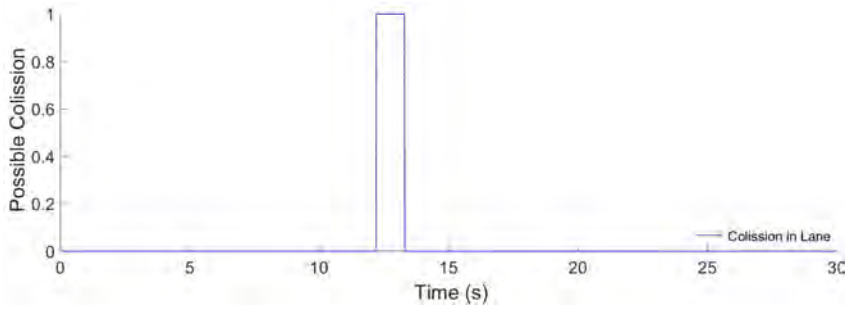
The performance of the maneuver, as illustrated in Figure 5.9, shows the detection of potential collision during the merging process, identified between 12 s and 13 s. MPC boundaries and the trajectory planning speed reference are displayed in Figure 5.9b. Here, the vehicle's slowing down at 11 s, braking at 13 s (Figure 5.9a), and subsequent acceleration at 15 s (once risk is eliminated) are observed.

Additional, Figures 5.9c and 5.9d present the vehicle's acceleration and jerk respectively. Both variables remain within the established limits, saturating in the limits of the MPC boundaries in case of the Jerk, thereby ensuring safe and comfortable planning for the driver during the merging process.

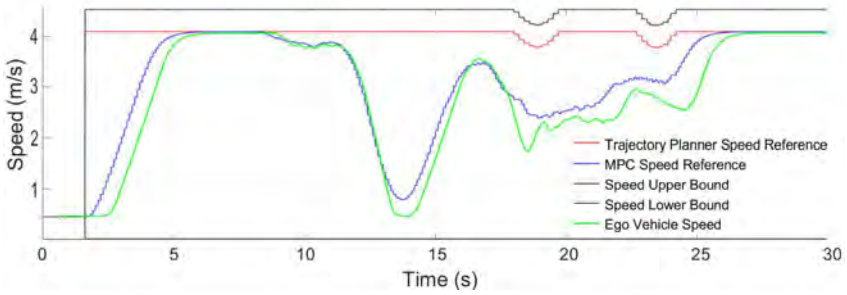
5.2.2 Roundabout Merging Mixed Test Environment Performance

The subsequent stage in the validation process for roundabout merging, following the established procedure within the ENABLE project [219], involves transitioning the tests into a more complex scenario that incorporates a mixed-test environment.

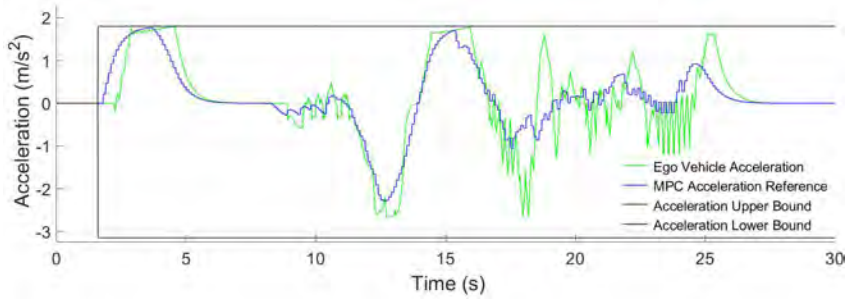
In this mixed environment, a real Renault Twizy is programmed to merge into a roundabout, while a virtual vehicle, simulated through the Dynacar simulator, was circulating within the roundabout.



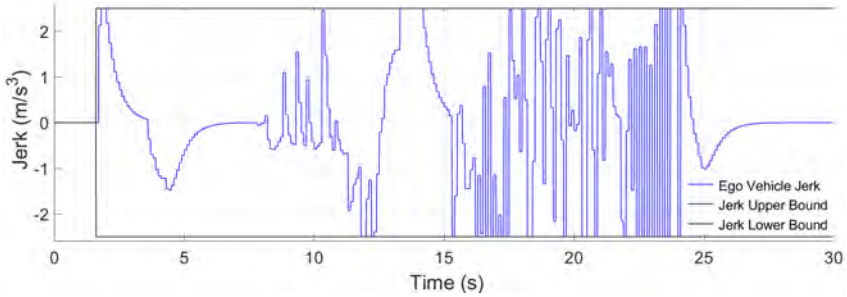
(a) Possible collision checker.



(b) Speed tracking performance with the MPC references and bounds



(c) Acceleration tracking performance with the MPC references and bounds.



(d) Jerk performance with the MPC references and bounds.

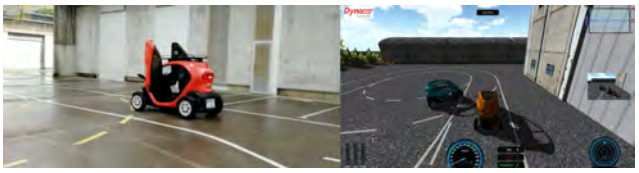
Fig. 5.9: Simulated performance of the merging vehicle using the HYTP.

Figure 5.10 presents a sequence of images elucidating the maneuver from the perspectives of both the real vehicle and Dynacar. Initially, the sequence depicts the vehicles in motion, with no imminent risk of collision (Figure 5.10a). Subsequently, the vehicle is shown decelerating (Figure 5.10b), and ultimately applying the brakes,

thereby conceding the right of way to the other vehicle in the roundabout (Figure 5.10c). The final images (Figures 5.10d and 5.10e) capture the vehicle as it circulates within and subsequently exits the roundabout.

(a) Vehicles before encounter ($t = 7\ s$).

(b) Real vehicle slowing down ($t = 10\ s$).



(c) Real vehicle stops at the entrance ($t = 13\ s$).

(d) Real vehicle circulating at the roundabout ($t = 16\ s$).

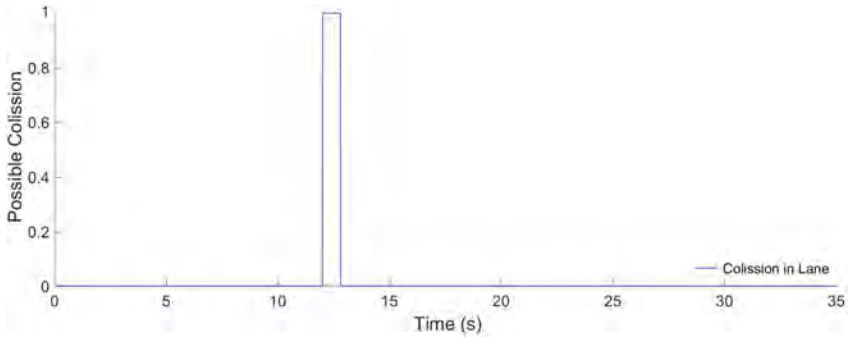
(e) Real vehicle exiting the roundabout ($t = 26\ s$).

Fig. 5.10: Images sequence of the merging at roundabout.

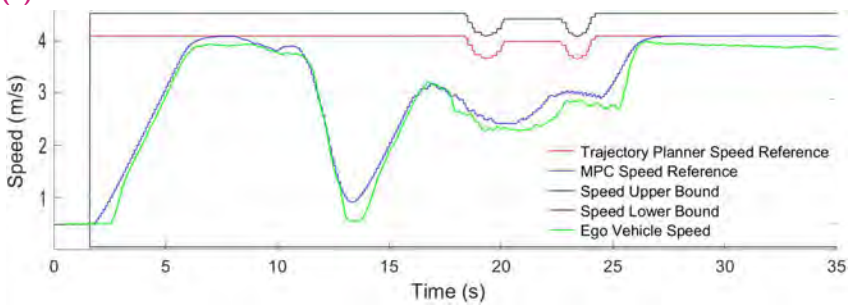
Figure 5.11 provides a comprehensive analysis of the maneuver performance. Beginning with Figure 5.11a, it illustrates the precise instance where the real vehicle identifies a potential collision risk with the virtual vehicle occurring around the second 12. Figure 5.11b demonstrates the real vehicle's speed, the MPC boundaries, and reference points. At 11 s, the vehicle begins to decelerate and continues until it applies the brakes at 13 s. Once the risk of collision is nullified at 14 s, the vehicle resumes acceleration. This sequence effectively illustrates the vehicle's ability to adhere to the reference throughout the entire planning process.

Furthermore, Figures 5.11c and 5.11d depict the vehicle's acceleration and jerk respectively. In certain instances, the vehicle surpasses the acceleration boundaries, a phenomenon attributed to an alternate actuator characterization that enables the vehicle to accelerate beyond the initially anticipated limits. Nonetheless, the jerk

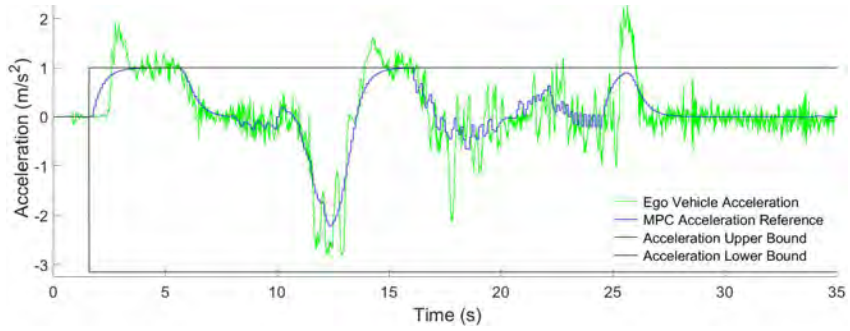
remains within the defined limits, ensuring a comfortable planning experience for the driver.



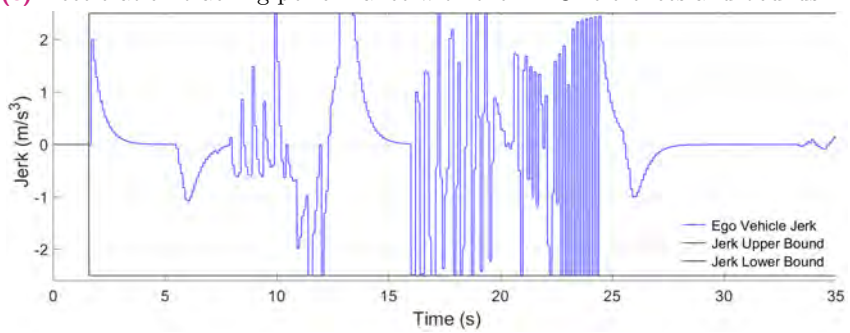
(a) Possible collision checker.



(b) Speed tracking performance with the MPC references and bounds



(c) Acceleration tracking performance with the MPC references and bounds.



(d) Jerk performance with the MPC references and bounds.

Fig. 5.11: Mixed environment performance of the roundabout merging, using the HYTP approach.

5.2.3 Platoon Lane Merging Simulated Performance

Upon successful implementation of the HYTP method for roundabout merging, further investigation into additional merging maneuvers involving multiple vehicles was conducted. This exploration focused on scenarios where vehicles needed to merge into another lane, making the lane merging scenario particularly intriguing. The interest in this scenario was heightened by considering situations involving platoons of vehicles, which offered a potential solution to the issue of low market penetration when vehicles drove in convoy formations.

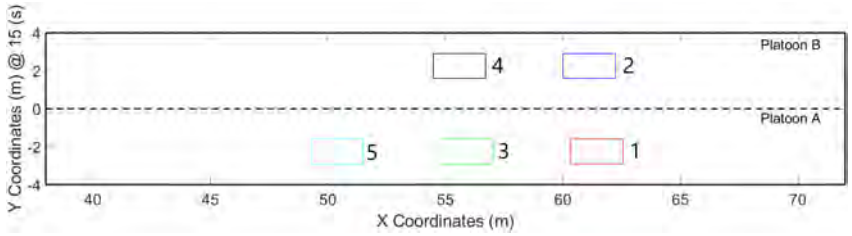
The research collaboration with the INRIA RITS team and the University of Berkeley provided the opportunity to further explore the lane merging maneuver. By combining the expertise in car following applications from both centers, this thesis was able to advance the development of the HYTP method. This collaborative effort made it feasible to integrate HYTP with car following controllers, leading to the approach described in Section 4.2.3.

The platoon lane merging maneuver was put into practice using the Dynacar simulator. The simulation involved five vehicles, modeled after the Twizy, with two on the left lane forming Platoon B and three on the right lane forming Platoon A. For the merging maneuver, the leader maintained a constant speed of 15 Km/h, simulating a low-speed platoon in an urban scenario.

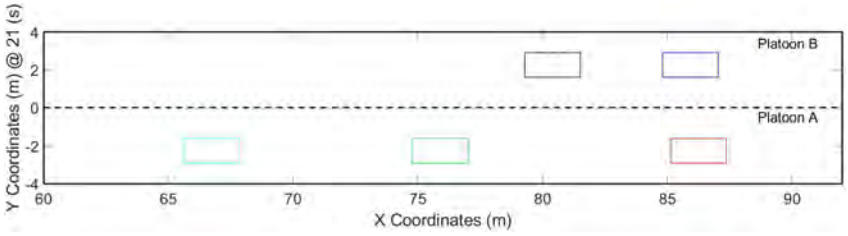
The process of the merge can be visualized through Figure 5.12, which provides an interval snapshot of the vehicle positions. At the 15 s, the merge request was sent, starting the process (Figure 5.12a). In the 21 s, Platoon A completed the task of opening a gap and sent the confirmation (Figure 5.12b). This action ensured that only a vehicle from Platoon B could merge when safe conditions were met, with the first condition being an adequate distance among the vehicles. In second 28, Platoon B finished adjusting its position according to the gap opened by Platoon A (Figure 5.12c). At the 33 s, Platoon B executed the lane change (Figure 5.12d), and by the second 39, Platoon A and Platoon B had successfully merged into a single Platoon C (Figure 5.12e).

Figure 5.13 demonstrates the longitudinal performance of both platoons during the merging process. Specifically, Figure 5.13a and Figure 5.13b present the vehicle speed of Platoon A and Platoon B, respectively. The distances between vehicles during the merging process are depicted in Figure 5.13c (Platoon A) and Figure 5.13d (Platoon B). Lastly, Figures 5.13e and 5.13f illustrate the time gap between the vehicles in Platoon A and Platoon B, respectively.

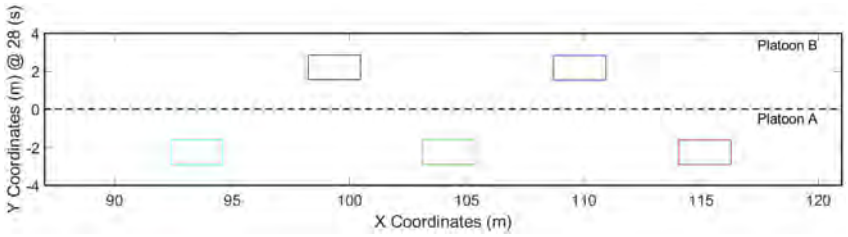
A closer examination of these figures reveals that at 15 s (as shown in Figure 5.13a), Platoon A decelerates to open a gap. Around the 23 s (shown in Figure 5.13b), Platoon B decelerates, aligning itself parallel to the space opened by Platoon A by using the projection of the vehicle leader of Platoon A as a reference. At the 28 s, the longitudinal reference switches to the MPC-generated one, facilitating the



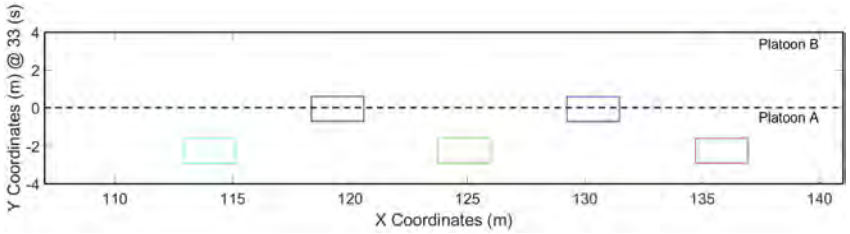
(a) Vehicles platoon before the maneuver begins ($t = 15\ s$).



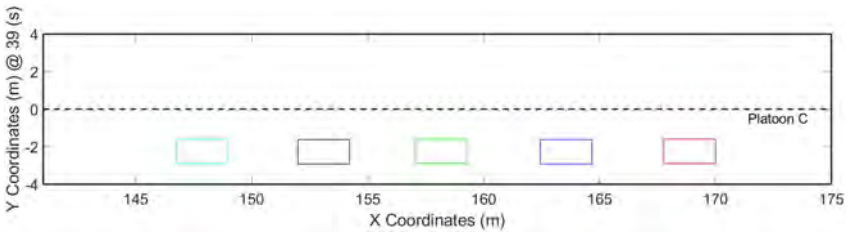
(b) Platoon A opening the gap ($t = 21\ s$).



(c) Platoon B adjusting its position ($t = 28\ s$).



(d) Platoon B changing lanes ($t = 33\ s$).

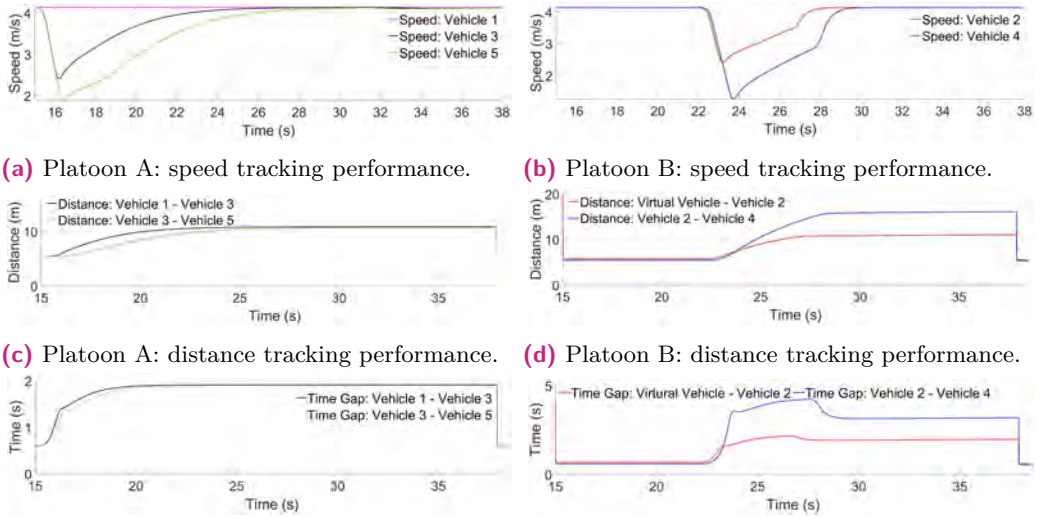


(e) Platoon C formed after Platoon B completed the lane change. ($t = 39\ s$).

Fig. 5.12: Simulated platoon lane merging using 5 vehicles with HYTP alongside the CACC.

execution of the lane change. Upon the completion of the lane change at the 39 s, Platoon B reverts to the CACC longitudinal reference, adjusting its position within the newly formed Platoon C.

As depicted in Figure 5.13c, the vehicles in Platoon A open a distance of 10 m, adhering to Equation 4.25. Concurrently, the positions in Platoon B are adjusted to align with the midpoint of the gap opened by Platoon A.



(a) Platoon A: speed tracking performance.

(b) Platoon B: speed tracking performance.

(c) Platoon A: distance tracking performance.

(d) Platoon B: distance tracking performance.

(e) Platoon A: time gap tracking performance.

(f) Platoon B: time gap tracking performance.

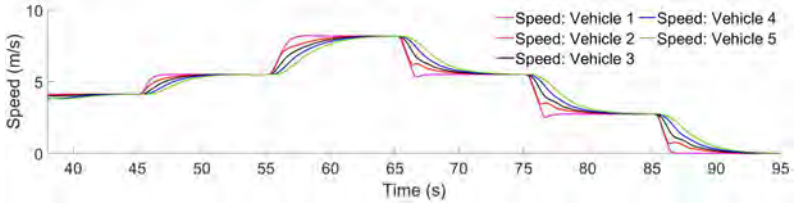
Fig. 5.13: Platoon A and Platoon B lane merging performance using the HYTP alongside the car following strategy using only CACC.

In terms of time gap (Figure 5.13e), it is observed that the gap between vehicles incrementally widens until the final value (h_{Final}) is reached, without any amplification. In Figure 5.13f, the vehicles in Platoon B are shown to increase the gap between them and the virtual vehicle until the lane change position is achieved. Following the completion of the lane change, the vehicles adopt the vehicles in front of them as references.

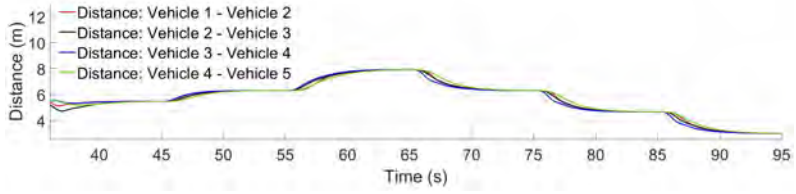
Upon completion of the merging process, the longitudinal performance of Platoon C is illustrated in Figure 5.14 from the moment the vehicles finalize the merge. Figure 5.14a represents the speed of the vehicles, Figure 5.14b depicts the distance between them, and Figure 5.14c shows the time gap among the vehicles. It is noteworthy that between seconds 39 and 45, as demonstrated in Figure 5.14a, the vehicles are observed closing the gaps. Despite this adjustment in positioning, indicating that vehicles did not merge into the anticipated position, the merge was successfully executed. Once the vehicles stabilized in the lane, various speed adjustments were introduced to demonstrate the effective performance of the newly formed Platoon C, which maintained the references without amplifying the changes. In Figure 5.14c, the sudden brake implemented by Vehicle 1 at second 85 causes the actual time gap between Vehicle 1 and Vehicle 2 to expand to infinity. This situation, predominantly due to the relation used to calculate the value, does not adversely impact the performance of the platoon. This is further evidenced by the smooth progression from their current speed to zero observed in the rest of the vehicles.

Figure 5.15 presents the lateral performance of Platoon B during the merging maneuver. Figure 5.15a illustrates the lateral references produced by the HYTP, while Figures 5.15b and 5.15c depict the angular and lateral errors, respectively. At the 26 s in Figure 5.15a, the lane change maneuver starts and the boundaries are

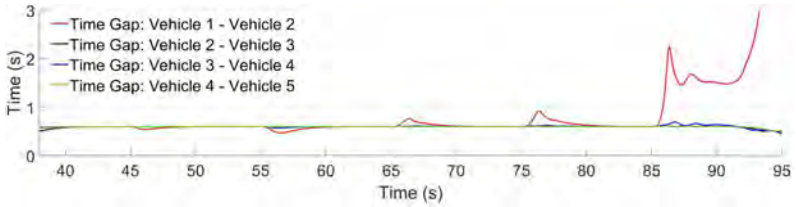
adjusted to accommodate the change in reference. Once Platoon B transitions to the other lane, the reference is reset to 0 m and begins to follow the trajectory of Platoon A. Notably, since both vehicles initiate the maneuver simultaneously and share identical dynamics, no discrepancy is observed in the execution of the lane change. Figures 5.15b and 5.15c display the angular and lateral errors respectively, with peak values recorded at 7.27^o and 4.5 m. The progression of both variables attests to the maneuver's comfort. However, the emphasis on comfort and safety during the maneuver results in a lane change process duration of 12 s, a time frame that is roughly average ranging from (1 s to 13.33 s) according to the literature [220].



(a) Platoon C: speed tracking performance.



(b) Platoon C: distance tracking performance

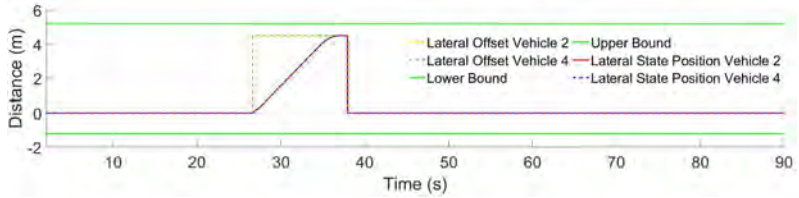


(c) Platoon C: time gap tracking performance.

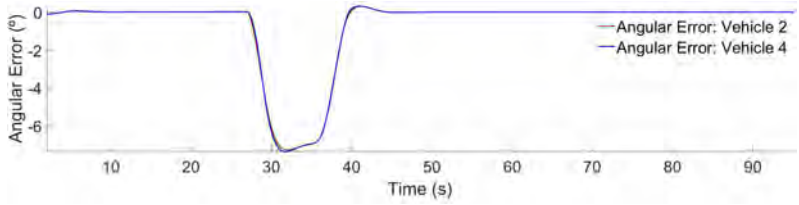
Fig. 5.14: Platoon C longitudinal performance using the HYTP alongside the car following strategy using only CACC.

5.2.4 Fleet Management and Smart Intersection Maneuvers Under the SerIoT Framework

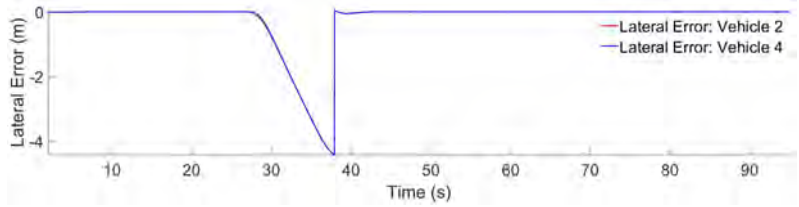
Within the context of the SerIoT project, two distinct maneuvers were selected. These maneuvers aim to demonstrate the system's proficiency in the field of CCAM, specifically within the framework of UC 2 - ITS in smart cities. The two maneuvers selected encompass fleet management and intersection management scenarios, each incorporating a unique detection and mitigation methodology. The application of the HYTP method in such maneuvers enhances its applicability across a broader range of cases. Notably, it ensures robust speed planning in instances necessitating the vehicle's stop at traffic lights or the prevention of potential collisions.



(a) Platoon B: lateral references, and performance of the MPC for the lane change manevuer.



(b) Platoon B: angular error of the lane change maneuver.



(c) Platoon B: lateral error of the lane change maneuver

Fig. 5.15: Lane change performance of the Platoon B using the HYTP alongside the car following strategy using only CACC.

The maneuvers were tested in a mixed environment, with a real vehicle executing the specific maneuver, and virtual components including vehicles and traffic lights supplementing the scenario. These tests were carried out at the Tecnalia test track using the Twizy as the real platform and the Dynacar simulator to virtualize the other vehicle. Traffic lights and the CS responsible for scenario supervision were simulated through MATLAB/Simulink software.

The **Fleet Management** scenario involves a fleet of vehicles tackling a traffic jam situation (Figure 5.16), where two possible routes are presented, the red one, (the first rout where the traffic jam is located) and the blue one (the alternative route). In this context, the route of Vehicle 2 (V2), which corresponds to the real one, is altered via V2I communications when Vehicle 1 (V1), the virtual vehicle, detects a traffic jam. This route modification is facilitated through communication with RSU 2 and the CS, which is tasked with vehicle monitoring. Throughout the maneuver, the vehicles send their respective routes to the CS using GN messages. When a traffic disruption is encountered, V1 sends a DENM to the CS as a warning. Upon receiving this alert, the CS dispatches a GN message indicating the revised route for V2 to follow. It should be noted that during these tests, the vehicles were operating at a maximum speed of 16 Km/h.

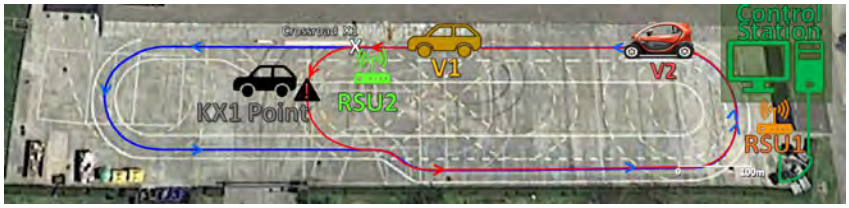
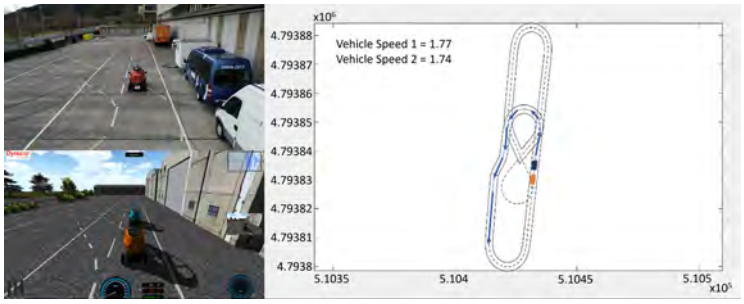
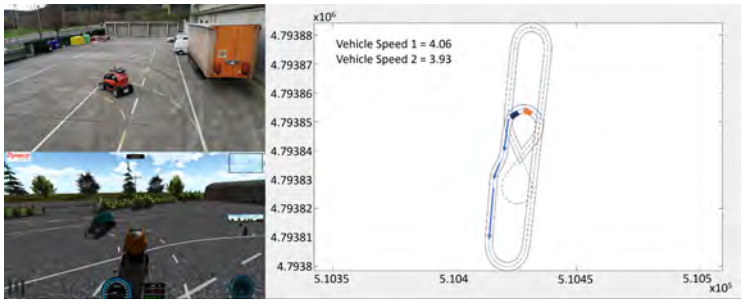


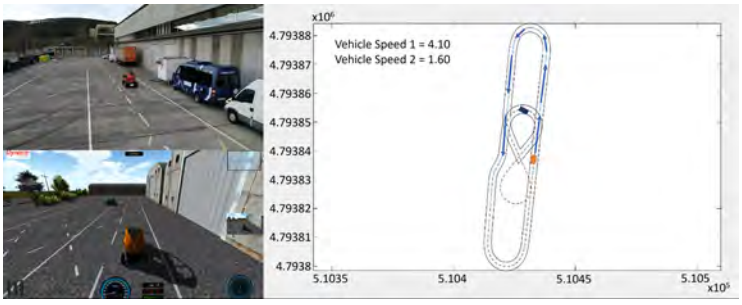
Fig. 5.16: Fleet Management scenario description within the SerIoT project [188].



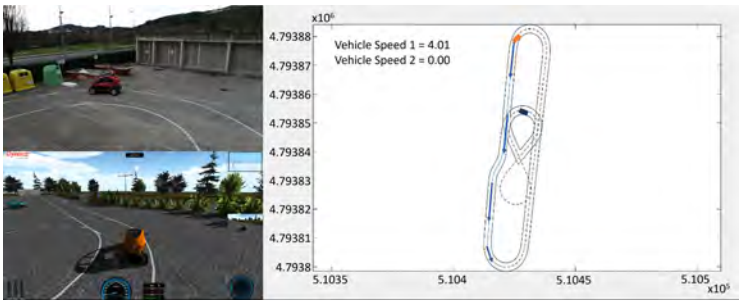
(a) Maneuver Start.



(b) Vehicle 1 following first route and continuing for a second lap.



(c) On second lap: Vehicle 2 changing to second route due to traffic stopping Vehicle 1.



(d) Vehicle 2 following second route.

Fig. 5.17: Fleet management maneuver sequence in traffic jam situation.

In order to evaluate the SerIoT system's robustness, a DoS attack was introduced into the infrastructure to disable the RSU2 in proximity to the conflict point (K1). This resulted in a shutdown of the RSU, preventing V2 from altering its route despite receiving a message from V1. Two mitigation strategies were proposed to counter the attack, however only the re-routing packets strategy was implemented [188].

Figure 5.17 presents a series of images that depict the positioning of both vehicles during the maneuver. The right figure plots the vehicles' positions over time, with the blue rectangle representing V1 and the orange rectangle representing V2. The lower left figure corresponds to the 3D visualization of the virtual vehicle in the Dynacar Visor, while the upper left figure represents the real vehicle. Figure 5.17a depicts both vehicles adhering to an identical path initially. Upon reaching the KX1 point during the second lap, V1 transmits a DENM message to the CS (Figure 5.17b). Subsequently, the CS instructs V2 to alter its route, as illustrated in Figure 5.17c. Figure 5.17d demonstrates V2 navigating through Route 2.

In the **Intersection Management** scenario, as depicted in Figure 5.18, Vehicle 1 (V1) follows a predetermined route and approaches an intersection. Concurrently, Vehicle 2 (V2) also follows a designated pathway and arrives at the same intersection from a different road segment. Traffic lights (TL1 and TL2) broadcast their respective information (state, time left before changing, etc.) via RSU1 and RSU2. V1 receives information from TL1 through its OBU and continues its trajectory if TL1 is green or stops if it is red. Similarly, V2 observes TL2 and its OBU to determine whether to continue or stop. The CS oversees the functioning of both traffic lights.

Fig. 5.18: Smart Intersection scenario description within the SerIoT project [188].

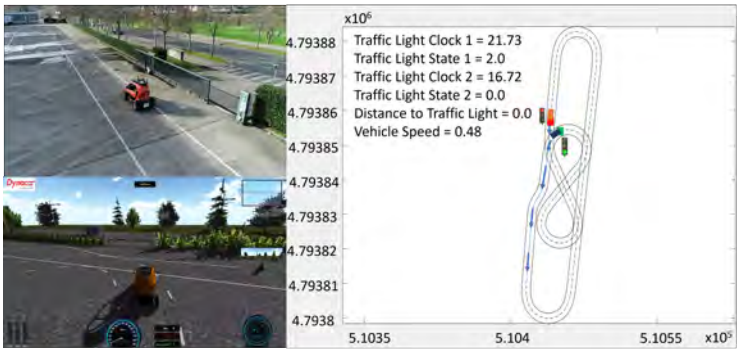
Similarly, as the fleet management scenario, a DoS attack was introduced into the infrastructure to test the resilience of the SerIoT system. This particularly attack aimed to induce a collision at the intersection by disabling RSU2, which was responsible for transmitting information regarding TL2. To mitigate this, a re-routing strategy was implemented. However, this scenario required the implementation of a distinct method for detecting and executing the re-routing.

Figure 5.19 shows an image sequence of the maneuver where it can be seen the vehicles crossing the intersection without difficulties, even though the DoS attack has occurred. Figure 5.19a presents the beginning of the scenario, where initially TL1 is in red and the TL2 in green. Figure 5.19b shows when V1 stops at the traffic light TL1, whereas V2 approaches TL2. In Figure 5.19c, it can be observed that

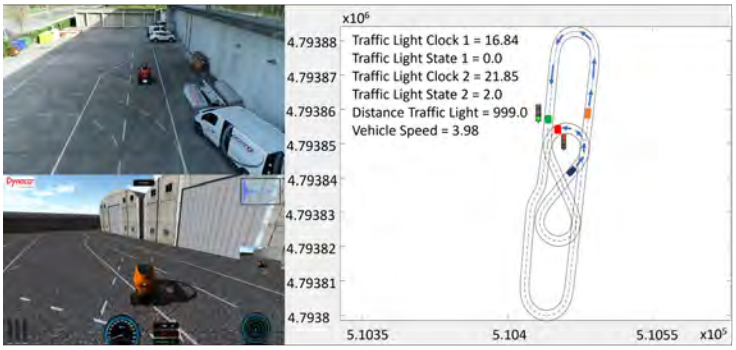
V2 is stopping at T2, while V1 accelerates due to TL1 changing to green. Finally, Figure 5.19d shows V2 accelerating once TL2 changes to green.

(a) Maneuver Start.

(b) V1 stopping at TL1.



(c) V1 accelerating and V2 stopping at TL2.



(d) V2 accelerating.

Fig. 5.19: Smart intersection maneuver sequence.

5.2.4.1 SerIoT System Performance

Table 5.3 illustrates the average reaction times of the SerIoT system under both scenarios, using evaluation metrics such as mitigation time, detection time, detected packet loss rate, and detection accuracy. The system exhibits a robust ability to detect any deviations from the standard network traffic pattern induced by the DoS attack in both instances, with an accuracy rate of 100 %. The first detection and mitigation method yields an average detection time of 4.34 s, while the second one achieves a detection time of 3.27 s, indicating a quicker detection with the latter method. The average time for the mitigation engine is 1.7 ms for the first method and 1.576 ms for the second, further demonstrating the enhanced speed of the second method in mitigating the attack.

Tab. 5.3: Performance of the SerIoT system in the fleet management and smart intersection scenarios.

Scenario	Mitigation Time	Detection Time	Detected Packet Loss	Accuracy of Detection
Fleet management	1.7 ms	4.34 s	0 %	100 %
Intersection management	1.576 ms	3.27 s	0 %	100 %

5.2.5 Discussions

This section evaluated the performance of the HYTP method in executing different cooperative maneuvers within both simulation and mixed test environments. The roundabout merging was validated, demonstrating the adaptability of the HYTP method beyond overtaking maneuvers. The tests, conducted in both simulated and mixed environments, provided satisfying results. The vehicle maintained safety and comfort by yielding at the entrance and proceeding once the risk was mitigated. Despite the real vehicle exceeding acceleration boundaries, it did not affect the maneuver's execution. However, this indicates the method's sensitivity to changes in the platform, an aspect that is undesirable at the decision stage.

The subsequent maneuver validated, was the platoon lane merging in simulation. This maneuver showed promising results as Platoon B was able to safely merge into the other lane. The decentralized method required low computational cost without depending on the platoon's leaders presenting low mean angular (0.6827^o) and lateral error (0.241 m). However, opportunities for improvement exist, particularly in reducing the time taken for the lane change execution (12 s), the overall maneuver execution time (25 s), and optimizing vehicle positioning post-lane change.

Regarding the fleet management and intersection management scenarios, the maneuvers were executed successfully with the aid of the SerIoT system. For the fleet management case, the vehicle adeptly altered its route in a timely manner to

avoid the traffic congestion. In the intersection management scenario, the vehicles demonstrated effective coordination by stopping at the intersection, thereby allowing the other vehicle to pass without any complications. Notably, irrespective of the detection and mitigation method employed, the SerIoT system delivered a response time less than 5 s. This is acceptable in vehicle traffic control contexts where non-critical communication is required.

The HYTP approach has demonstrated satisfactory performance overall. It executed each maneuver safely and comfortably, contributing valuable insights into the execution of merging maneuvers. Nevertheless, the absence of a distinct behavioral planner posed a challenge during maneuver testing, as the method needed pre-configuration for each maneuver, leading to extended testing time. This issue could also impact performance, given the necessity for accurate MPC configuration. Consequently, the RTTP emerges as a possible solution to these challenges while simultaneously aiming to enhance the performance of the maneuvers. The results of this approach will be discussed in the following section.

5.3 Real Time Trajectory Planning

In view of the limitations identified with the HYTP approach, this study delves into an alternative decision method, namely the RTTP. Elaborated in Section 4.3, this method with a detailed map information, integrates a FSM to allow the vehicle to respond to various scenarios, and incorporates real-time trajectory generation. It is coupled with the car following controller for platoon applications, particularly for platoon lane merging maneuvers. With this approach, three significant milestones were achieved. The subsequent enumeration provides a detailed explication of each. validating test.

1. The combination of the RTTP decision method with the car following controller for platoon lane merging maneuvers in a simulated urban environment, representing its successful integration.
2. The execution of RTTP for platoon lane merging in mixed test environments provided evidence of the strategy's applicability in real-world conditions.
3. Multiple executions of the RTTP for platoon lane merging in mixed test environments, where the merging vehicle incorporates from different positions to help corroborate the functionality of the IoTAC system in the CCAM field.

The following sections will present the results corresponding to each test, accompanied by detailed descriptions of the methodologies employed in their execution.

5.3.1 Platoon Lane Merging Simulated Performance

The platoon lane merging maneuver, tested with the RTTP approach, is simulated using ROS + CARLA. Six vehicles, modeled on the Renault Twizy 80, were incorporated into the simulation. Specifically, three vehicles on the left lane consti-

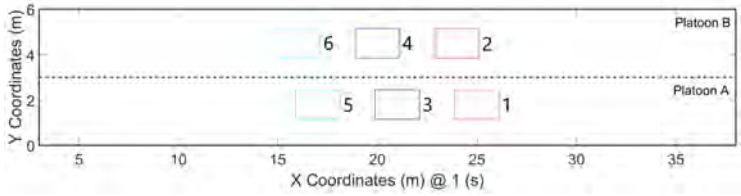
tuted Platoon B, while the other three vehicles on the right lane formed Platoon A. Throughout the merging process, the vehicles maintained a constant speed of 18 Km/h. Following the completion of the maneuver, the speed of the leading vehicle was varied to assess the performance of the newly established platoon under different conditions. These tests incorporated a simulated delay of 100 ms.

These tests were executed as part of Use Case 2.1 - Advanced Platoon Operations - within the scope of the AUTOEV@l project, which focuses on platoon applications in urban scenarios. Tecnalia's vicinity was selected as the testing environment due to its relevance for the maneuver. The aim of these tests is to transition this maneuver into more realistic settings.

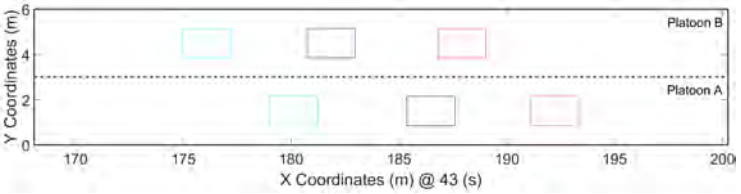
Figure 5.20 provides a time-sequenced visual representation of the vehicles' positions, demonstrating a seamless transition from Platoon B to Platoon A. The process, as detailed in Section 4.1, initiates with the vehicles operating in automated mode (as seen in Figure 5.20a). At 24 s, a signal prompts the formation of Platoon A and Platoon B, marking the beginning of the procedure. By the second 43, the start of the lane merging process is observed, triggered by Platoon B's merge request upon detecting a lane blockage (Figure 5.20b). Upon receipt of the request, Platoon A conducts a situational assessment, and, deeming conditions safe, accedes to the request. This decision culminates in both platoons concurrently adjusting their positions until the 61 s (Figure 5.20c). Then, the lane change is executed by Platoon B, which is visible at second 63 in Figure 5.20d and concludes by the second 64. Ultimately, at the second 83, the newly formed Platoon C is visible (Figure 5.20e).

Figure 5.21 illustrates the longitudinal performance and the FSM state transitions of both platoons during the merging process. Specifically, Figure 5.21a represents the vehicle speed of Platoon A, while Figure 5.21b depicts the vehicle speed of Platoon B. The distance between vehicles during the merging process is captured in Figures 5.21c and 5.21d, corresponding to Platoon A and Platoon B respectively. Figures 5.21e and 5.21f shoes the time gap between the vehicles in Platoon A and Platoon B. Lastly, Figures 5.21g and 5.21h presents the state transitions during the maneuver.

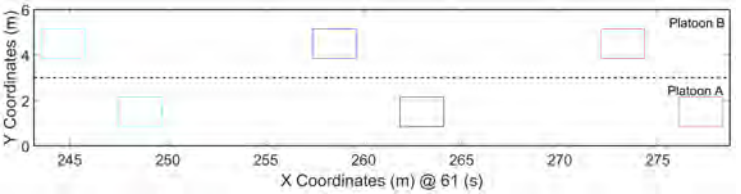
Upon a detailed examination of Figures 5.21a and 5.21b, it is observed that both platoons begin to form at the 24 s, with Platoon A and Platoon B establishing themselves within 5 s and 2 s respectively. At second 43, Platoon A initiates the gap opening process, while Platoon B commences adjusting its position parallel to the space unveiled by Platoon A. Both maneuvers are executed without amplifications. In the second 61 s, once the gaps are completely opened, Platoon B initiates the lane change while maintaining the speed of Platoon A's leader. The maneuver is completed at the second 64, where Platoon B reverts to the car following control and adjusts its position within the newly formed Platoon C.



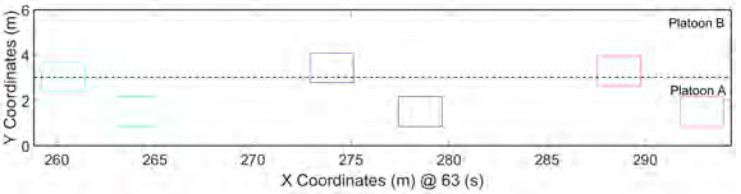
(a) Vehicles before forming Platoon A and B ($t = 1\ s$).



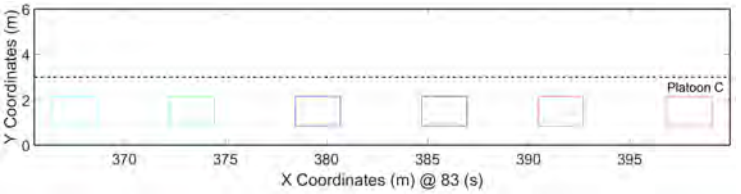
(b) Vehicles platoon before the maneuver begins ($t = 43\ s$).



(c) Platoon A and B finish adjusting their position ($t = 61\ s$).



(d) Platoon B changing lanes ($t = 63\ s$).

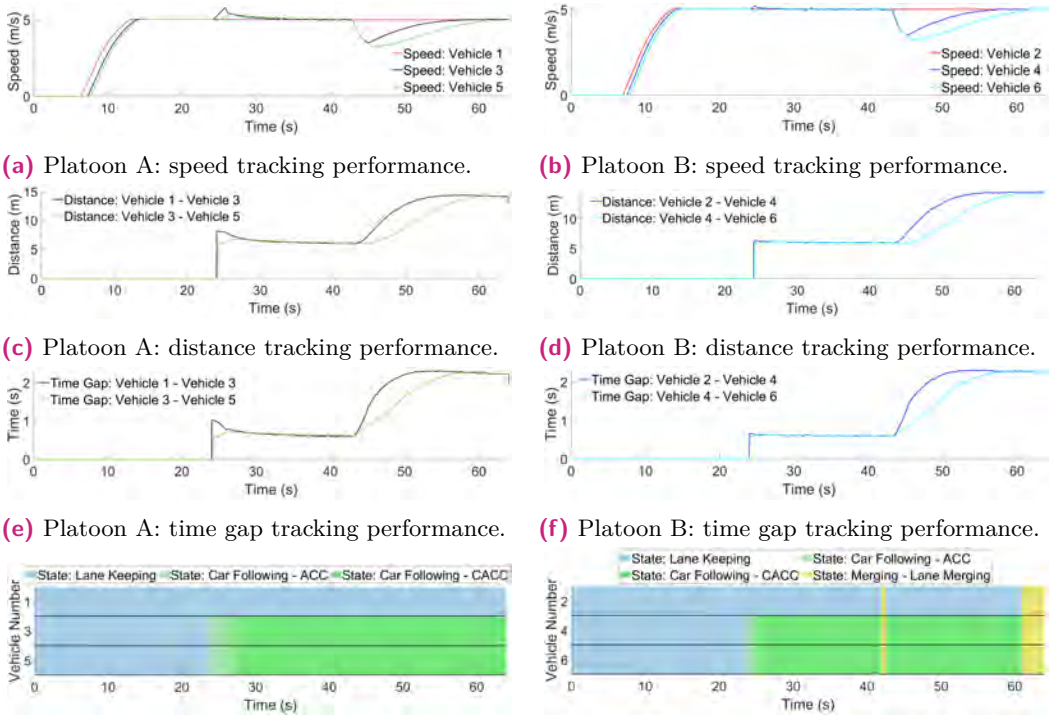


(e) Platoon C formed after Platoon B completed the lane change. ($t = 83\ s$).

Fig. 5.20: Images sequence of the simulated platoon lane merging using 6 vehicles with RTTP approach alongside the car following strategy.

Figure 5.13c reveals that the vehicles in Platoon A open a distance of 14 m in accordance with Equation 4.25, while the positions in Platoon B are adjusted based on the midpoint of the gap opened by Platoon A.

In terms of time gap, Figure 5.21e shows that the gap between vehicles in Platoon A expands until the final value h_{Final} is reached, without any amplification. Conversely, Figure 5.13f demonstrates that the vehicles in Platoon B increase the gap between themselves and the virtual vehicle until the lane change position is achieved. Subsequent to the completion of the lane change, the vehicles adopt the predecessor in front of them as references.



(a) Platoon A: speed tracking performance.

(b) Platoon B: speed tracking performance.

(c) Platoon A: distance tracking performance.

(d) Platoon B: distance tracking performance.

(e) Platoon A: time gap tracking performance.

(f) Platoon B: time gap tracking performance.

(g) Platoon A: FSM states of each vehicle.

(h) Platoon B: FSM states of each vehicle.

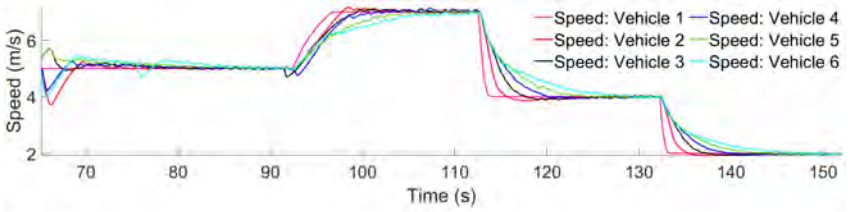
Fig. 5.21: Simulated longitudinal performance of Platoon A and B during the platoon lane merging, using the RTTP approach alongside the car following strategy.

The process can be seen as well in Figure 5.21h, were the states are observed, first in *lane keeping*, then in *car following*, when the lane blockage is detected for a few iterations the state changes to *merging*, where is decided that a position adjustment is needed so it transitions again to the *car following* state where open gap operation is executed until reaching the merging point and changing back to the the *merging* state to execute the lane change.

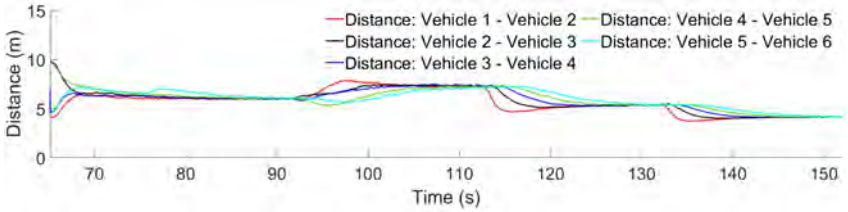
In case of the Figure 5.21g, it is only observe the transition from *lane keeping* state to *car following*, since they don't execute other type of maneuver.

Upon successful completion of the merging procedure, Figure 5.22 illustrates the longitudinal performance of Platoon C from the moment the vehicles finalized their merging. Figure 5.22a presents the speed of the vehicles, while Figure 5.22b displays the inter-vehicle distance, Figure 5.22c portrays the time gap among them, and Figure 5.22d displays the state transitions of each vehicle.

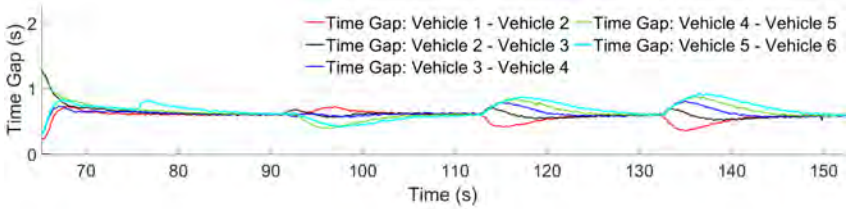
Notably, between seconds 64 and 69, as exhibited in Figure 5.22a, the vehicles are observed to gradually close the gaps until they reach their respective references at second 84. Since the vehicles are already in the platoon, they immediately change to *car following* state, more precisely to CACC. Despite the successful completion of the merge, the necessity for post-merging adjustments indicates that the vehicles did not precisely meet the anticipated positioning. This can be attributed to the vehicles in Platoon B executing the lane change as soon as sufficient space is perceived, a strategy



(a) Platoon C: speed tracking performance.



(b) Platoon C: distance tracking performance



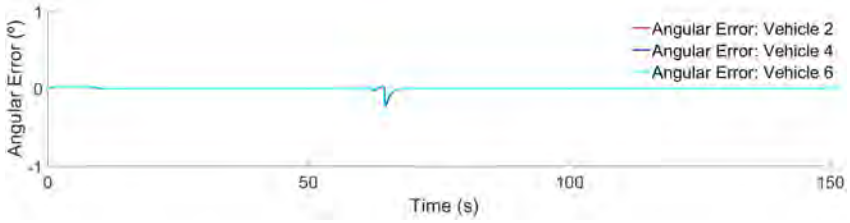
(c) Platoon C: time gap tracking performance.

(d) Platoon C: FSM states of each vehicle.

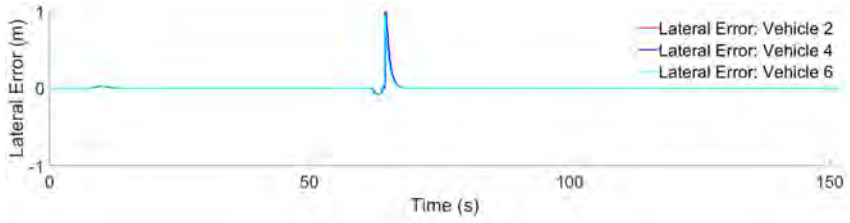
Fig. 5.22: Simulated longitudinal performance of the Platoon C during the platoon lane merging, using the real time trajectory approach alongside the car following strategy.

intended to minimize the maneuver time, as well as the properer configuration of the point where the lane merge must be executed. Once the vehicles achieved stability in their lane, various speed alterations were introduced to demonstrate the performance of the newly-formed Platoon C, effectively maintaining the references without amplifying the changes.

Figure 5.23 presents the lateral performance of Platoon B during the merging maneuver, specifically illustrating the angular error and lateral error in Figures 5.23a and 5.23b respectively. The simultaneous execution of the maneuver Platoon B vehicles, attributable to their identical dynamics, results in an indistinguishable lane change execution. Angular error and lateral error presents peak values of -0.25^o and 1 m respectively. Importantly, the progression of both variables signifies the comfort and swiftness of the maneuver, with the lane change process being completed in a mere 3 s.



(a) Platoon B: angular error of the lane change maneuver.



(b) Platoon B: lateral error of the lane change maneuver

Fig. 5.23: Platoon B simulated lateral performance during platoon lane merging, using the RTTP approach alongside the car following strategy.

5.3.2 Case 1: Platoon Lane Merging Mixed Environment Performance

As a part of the validation process for this maneuver within the scope of the AUTOEV@l project, and as a base maneuver for testing the IoTAC system, a mixed test environment is adopted for transitioning this maneuver to a real-world scenario. This environment comprises three vehicles: one executing the merging operation and two forming a platoon to accommodate the merging vehicle.

The selected vehicles included a real Twizy, a vehicle simulated solely in ROS using the Twizy kinematic model, and a third vehicle simulated using a combination of ROS and CARLA, employing a model similar to the Renault Twizy 80. In this scenario, the two virtual vehicles were positioned in the right lane, forming a two-vehicle platoon, while the Twizy was positioned in the left lane with the objective of executing the lane merging operation. The tests were conducted at the Tecnalia Test Track. A maximum speed of $2m/s$ was set for the lead vehicle.

Figure 5.24 provides a visual sequence of the maneuver from two distinct perspectives. The left images represent the perspective of the Twizy, while the right images illustrate the CARLA perspective, where the operations of the three vehicles are monitored. Initially, Figure 5.24a displays the merging vehicle operating in automated mode with the two virtual vehicles forming Platoon A. Subsequently, Figure 5.24b presents the vehicles prior to the beginning of the merging process. In Figure 5.24c, Platoon A is depicted with an opened gap, ready for the merging vehicle to execute the lane change. The lane change process is captured halfway in Figure 5.24d, and the final image, Figure 5.24e, shows the Platoon B.

Figure 5.25 provides a comprehensive overview of the longitudinal performance of Platoon A and the merging vehicle during the maneuver. More specifically, the speed

(a) Merging Vehicle and Platoon A ($t = 10\ s$).

(b) Vehicles platoon before the maneuver begins ($t = 18\ s$).

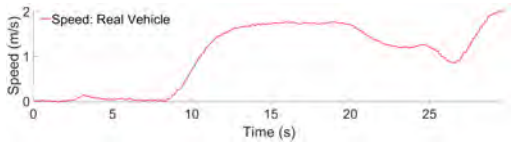
(c) Platoon A and Merging Vehicle with the position adjusted, before the lane change ($t = 21\ s$).

(d) Merging Vehicle changing lanes ($t = 22\ s$).

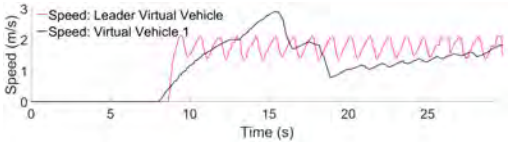
(e) Platoon B formed after Merging Vehicle completed the lane change. ($t = 27\ s$).

Fig. 5.24: Images sequence of the mix test environment platoon lane merging using 3 vehicles with RTTP approach alongside the car following strategy.

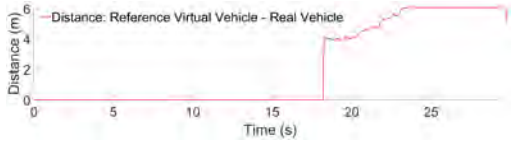
of the merging vehicle is depicted in Figure 5.25a, while Figure 5.25b illustrates the speed of the vehicles in Platoon A. The distances among these vehicles during the merging process are represented in Figures 5.25c and 5.25d, for the merging vehicle and Platoon A, respectively. Figures 5.25e and 5.25f portray the time gap between the merging vehicle and the leading vehicle's virtual projection, as well as that of Platoon A, respectively. In Figures 5.25g and 5.25h the FSM states of the merging vehicle and Platoon A are observed respectively.



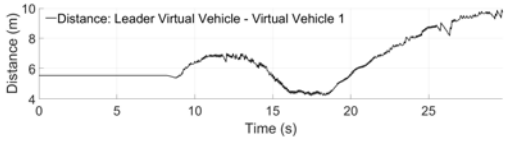
(a) Merging Vehicle: speed tracking performance.



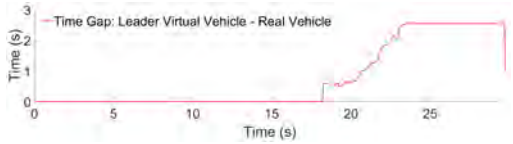
(b) Platoon A: speed tracking performance.



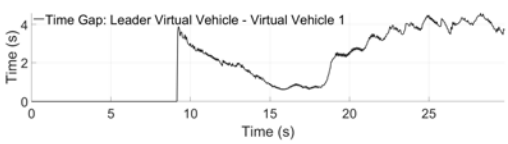
(c) Merging Vehicle: distance tracking performance.



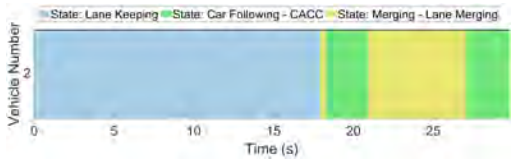
(d) Platoon A: distance tracking performance.



(e) Merging Vehicle: time gap tracking performance.



(f) Platoon A: time gap tracking performance.



(g) Merging Vehicle: FSM states of the real vehicle.

(h) Platoon A: FSM states of each vehicle.

Fig. 5.25: Longitudinal performance of Platoon A and the merging vehicle during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy.

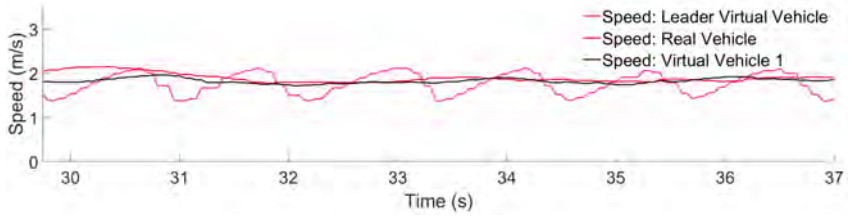
A detailed analysis of these graphs reveals that the merging vehicle operates in *lane keeping* state, as seen in Figures 5.25g and 5.25h. The virtual vehicles initiate the formation of Platoon A at second 9 switching to *car following* state and at the second 16 is fully attached to the platoon, running in CACC. At the second 18, the platoon lane merging starts with Platoon A opening the gap. Simultaneously, the merging vehicle adjusts its position in parallel to the space created by Platoon A. At the second 21, the gap is fully opened and the merging vehicle initiates the lane change process, maintaining the speed of the Platoon A leader. The maneuver concludes in the 27 s, as the merging vehicle switches back to the *car following* state and adjusts its position in the newly formed Platoon C.

These transitions can be observed in Figures 5.25a and 5.25b, which illustrate the necessary speed adjustments required to execute the maneuver. Specifically, the merging vehicle slows down to align its position, while the virtual follower vehicle also reduces its speed after is fully integrated with Platoon A. Notably, the vehicle simulated in CARLA exhibits instability at speeds lower than 2.77 m/s, a phenomenon attributed to the inaccuracies in the model used. As depicted in Figure 5.13c, the vehicles in Platoon A create a distance of 10 m in accordance with

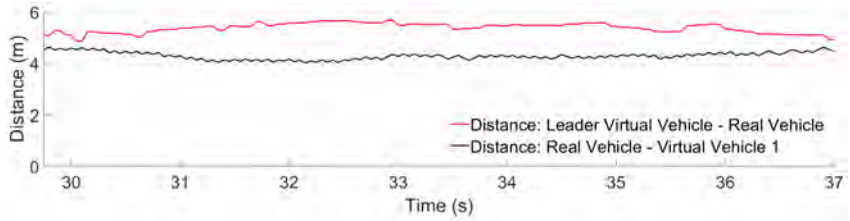
Equation 4.25, while the position of the merging vehicle adjusts to align with the midpoint of the gap created by Platoon A.

In terms of the time gap, Figure 5.21e shows that the gap between vehicles increased until the final value, h_{Final}, was reached. Figure 5.13f demonstrates that the merging vehicles increased the gap between themselves and the virtual leader vehicle until the lane change position is achieved. Upon completion of the lane change, the vehicles adopted the predecessor of them as references.

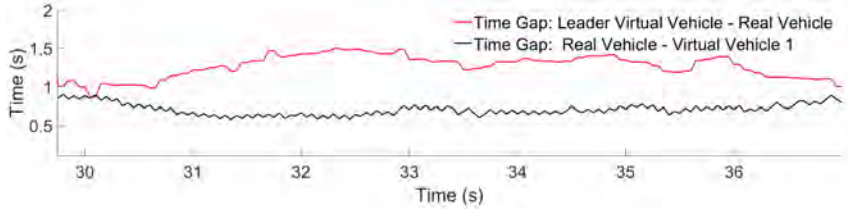
Figure 5.26, illustrate the longitudinal performance of Platoon B post-merging. Figure 5.26a presents the vehicle speed, Figure 5.26b highlights the inter-vehicle distance, Figure 5.26c depicts the time gap between vehicles, and Figure 5.26d shows the FSM sates of each vehicle.



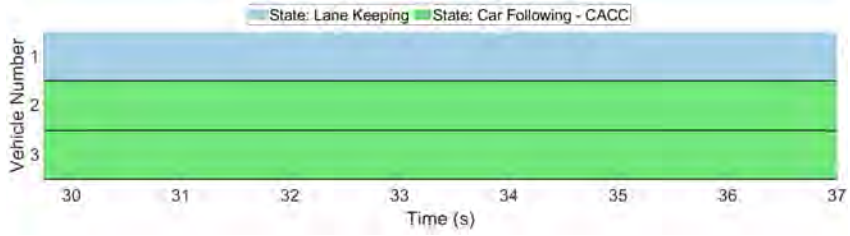
(a) Platoon B: speed tracking performance.



(b) Platoon B: distance tracking performance



(c) Platoon B: time gap tracking performance.



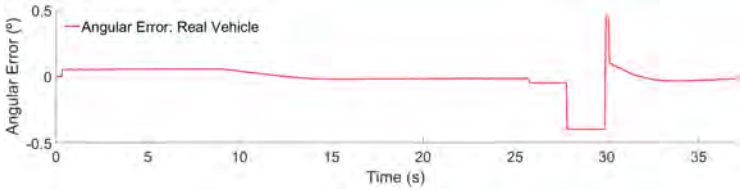
(d) Platoon B: FSM states of each vehicle.

Fig. 5.26: Platoon B mixed test environment longitudinal performance during the platoon lane merging, using the RTTP alongside the car following strategy.

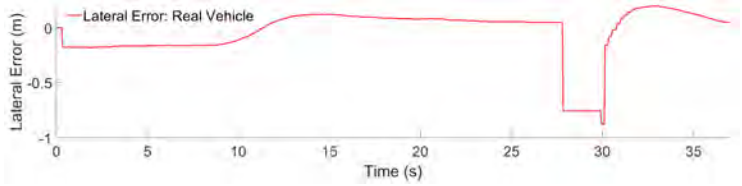
Notably, despite the CARLA-simulated vehicle demonstrating some instability at low speeds, all vehicles in Platoon B maintained a consistent speed throughout the test. According to Figure 5.26b, the real vehicle successfully merged into the

platoon, maintaining a distance of 5 m from the leader and 4 m from the vehicle behind, resulting in an average time gap of 1.3 s and 0.7 s respectively, as shown in Figure 5.26c. While the initial follower displayed a slightly increased error, it was effectively attenuated by the third vehicle, preventing propagation.

The lateral performance of the merging vehicle is depicted in Figure 5.27, with the angular error and lateral error illustrated in Figures 5.27a and 5.27b, respectively. The maximum values noted were -0.39^o and -0.87 m. Moreover, the progression of both variables suggests that the maneuver was conducted smoothly, with the lane change process finalizing within approximately 3 s, mirroring the simulation results. However, once the vehicle enters the main lane, it necessitates an adjustment in its lateral position. This issue could be mitigated by allowing more time for the lane change, thereby ensuring an even smoother progression.



(a) Merging Vehicle: angular error of the lane change maneuver.



(b) Merging Vehicle: lateral error of the lane change maneuver

Fig. 5.27: Merging vehicle Lateral performance during the platoon lane merging, using the RTTP approach alongside the car following strategy in a mixed test environment.

5.3.3 Case 2: Platoon Lane Merging with the IoTAC Framework

In a manner similar to the SerIoT project, the IoTAC project also aims to evaluate their proposed solutions in various real-life scenarios, specifically in the domain of CCAM. Consequently, multiple tests were conducted under the project to evaluate the performance of the platoon lane merging maneuver from different positions. This included the previously depicted scenario as well as a new case where the merging occurs from behind. These tests employ a mixed-environment configuration, where the Twizy performs the lane change, the leader is simulated through the CARLA model, and the follower is represented by the kinematic bicycle model as a virtual vehicle. Notably, these tests incorporate both the CS and the IoTAC modules for a comprehensive assessment.

As mentioned in Section 3.1.2, the Twizy is equipped with a KSG and an attack detection module, both of which constantly monitored the network traffic

of the platform. Moreover, to ensure robust performance, the CS is equipped with additional IoTAC modules, such as the Honeypot, the RMS, and the FEAM. These are actively engaged in tasks such as monitoring and implementing actions.

In the context of this maneuver, there is continuous communication via MQTT between the vehicles and the CS. The CS, upon detecting that the Twizy is operating on a closed lane, sends a notification prompting the vehicle to change lanes and form a platoon. Upon receipt of this notification, the Twizy initiates the negotiation process to change lanes and form the platoon.

During the execution of the maneuvers, two types of cyber attacks were simulated: a Port Scan and a DoS. These simulations were conducted with the dual objectives of verifying the correct detection of these attacks, and assessing whether the presence and activities of these modules could potentially impact the performance of the maneuvers.

Figure 5.28 presents a sequence of images that capture different stages of the maneuver. The images on the left depict the perspective of the Twizy, while the images on the right demonstrate the perspective from CARLA, where three vehicles are under observation. As illustrated in Figure 5.28a, the vehicle to merge is seen operating in automated mode, while the two virtual vehicles have already formed Platoon A. Figure 5.28b captures the vehicles just before the initiation of the merging process. The vehicle in the midst of the lane change process is depicted in Figure 5.28c, and finally, Figure 5.28d showcases the successful formation of Platoon B.

Figure 5.29 presents an analysis of the longitudinal performance of Platoon A and the merging vehicle during the maneuver. Specifically, Figure 5.29a illustrates the speed of the merging vehicle, while Figure 5.29b depicts the speed of the vehicles within Platoon A. Additionally, Figures 5.29c and 5.29d represent the inter-vehicle distance during the maneuver for the merging vehicle and Platoon A, respectively. Figures 5.29e and 5.29f provide the time gap between the merging vehicle and the leader vehicle's virtual protection, as well as within Platoon A. Lastly, Figures 5.29g and 5.29h presents the FSM sates of the merging vehicle and the Platoon A respectively.

An in-depth analysis of Figures 5.29g and 5.29h reveals that the vehicle executing the merge operated in *lane keeping* state, while the virtual vehicles began to form Platoon A in the second 5, being fully integrated in the platoon at second 11. Since, no position adjustment is necessary, at second 13 the merging vehicle proceeds to *merging* state and execute the lane change. at second 19 the merging vehicle switch to *car following* state, being fully attached to the platoon.

From Figures 5.29a and 5.29b it can be observed that during the merging process the vehicles in Platoon A maintained their speeds, and the merging vehicle had a minor variance ($\pm 0.4\ m/s$) in its speed during the lane change.

As depicted in Figure 5.29c, during the lane change at the second 19, the merging vehicle initiated the calculation of the distance to the first follower virtual vehicle.

(a) Merging Vehicle and Platoon A ($t = 1\ s$).

(b) Vehicles platoon before the maneuver begins ($t = 13\ s$).

(c) Merging Vehicle changing lanes ($t = 15\ s$).

(d) Platoon B formed after Merging Vehicle completed the lane change. ($t = 19\ s$).

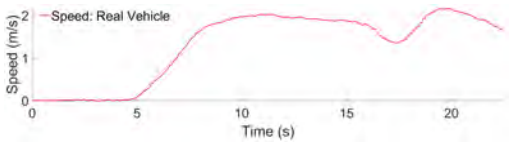
Fig. 5.28: Mixed test environment platoon lane merging test scenario using three vehicles, with RTTP and the car following strategy under the IoTAC project framework.

This suggests that the vehicle was nearing its target lane. Figure 5.29d reflects how the first virtual vehicle decreased the distance to the leading virtual vehicle, achieving a final distance of $4\ m$.

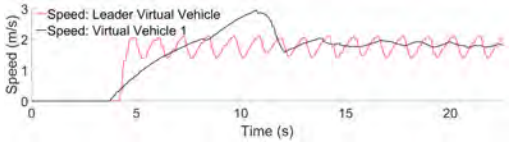
In Figure 5.29e, the merging vehicle is observed to increase the time gap as it adjusts its position in the lane. Correspondingly, Figure 5.29f displays a similar behavior in terms of distance, with the vehicle reducing the time gap until it reaches a value of $0.6\ s$.

Upon completion of the merge, the longitudinal performance of Platoon B is presented in Figure 5.30c, from the point where the vehicles completed the merging process. Figure 5.30a illustrates the speed of the vehicles, Figure 5.30b presents the distance between them, Figure 5.30c demonstrates the time gap among the vehicles, and Figure 5.30d represents the FSM states of each vehicle.

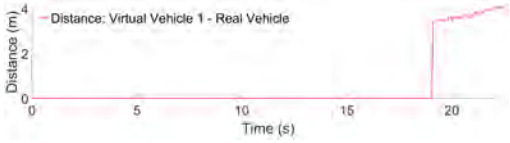
Figure 5.30a displays the vehicles operating within Platoon B. Throughout this test, the vehicles altered their speed up to $2.77\ m/s$. Specifically, the vehicle simulated



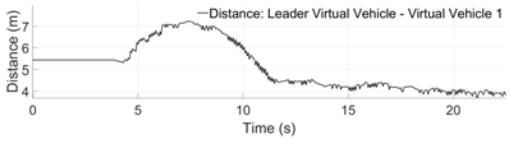
(a) Merging Vehicle: speed tracking performance.



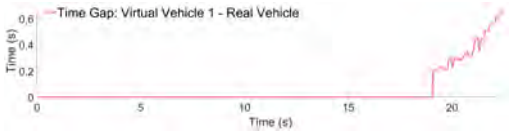
(b) Platoon A: speed tracking performance.



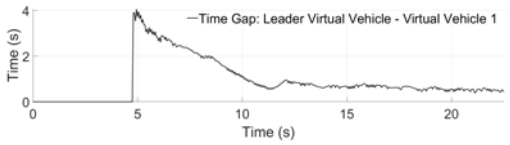
(c) Merging Vehicle: distance tracking performance.



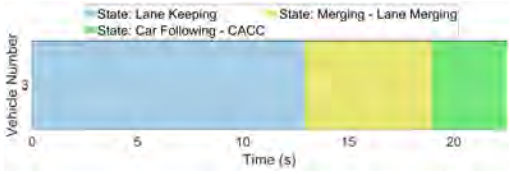
(d) Platoon A: distance tracking performance.



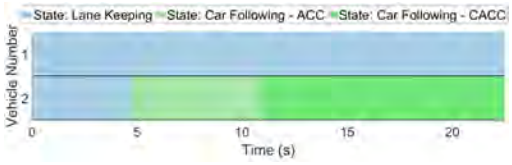
(e) Merging Vehicle: time gap tracking performance.



(f) Platoon A: time gap tracking performance.



(g) Merging Vehicle: FSM states of the real vehicle.



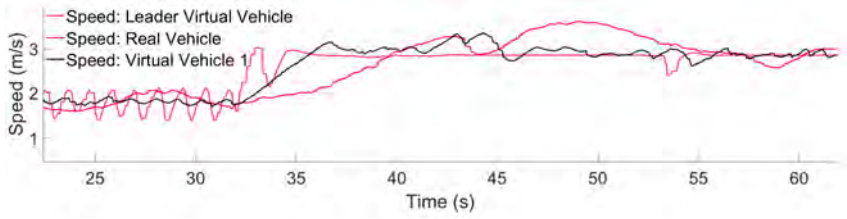
(h) Platoon A: FSM states of each vehicle.

Fig. 5.29: Longitudinal performance of Platoon A and the merging vehicle during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy.

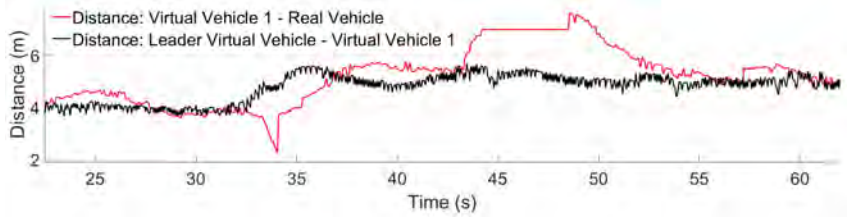
in CARLA demonstrated stable behavior during speed increase, maintaining a steady speed of 2.7 m/s. Conversely, the real vehicle momentarily exceeded the anticipated speed due to a nearby tight curve segment that necessitated additional adjustments. This phenomenon is evident in Figure 5.30b, which shows a 7 m distance for the real vehicle as opposed to a 5 m distance for the virtual follower. Concerning the position of the merging vehicle upon arrival, it initially held a distance of 4.6 m before swiftly adjusting. In terms of time gap, the merging vehicle and the virtual vehicle maintained average time gaps of 0.7 s and 0.6 s respectively.

The Figure 5.31, exhibits the lateral performance of the vehicle whilst merging. Specifically, Figures 5.31a and 5.31b represent the angular error and lateral error, respectively. The peak values recorded are 0.49^o degrees for angular error and -0.45 m for lateral error. Furthermore, the progression of both variables indicates not only the comfort of the maneuver but also its rapid execution, with the lane change process being completed in a span of 3 s. However, in the same way as the previous tests, the vehicle needed a lateral adjustment once finished the lane change.

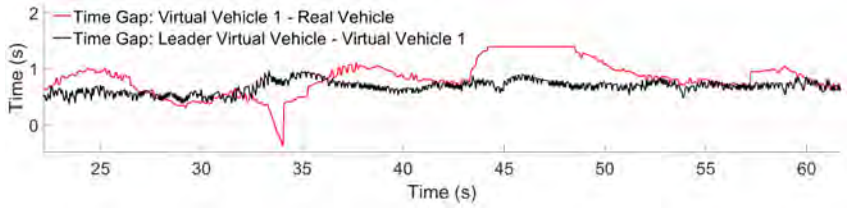
The modules associated with IoTAC installed in the vehicle demonstrated a high level of effectiveness. Specifically, the attack detection module had a success rate of



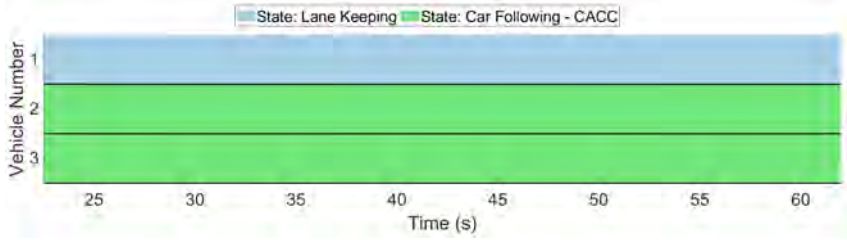
(a) Platoon B: speed tracking performance.



(b) Platoon B: distance tracking performance



(c) Platoon B: time gap tracking performance.

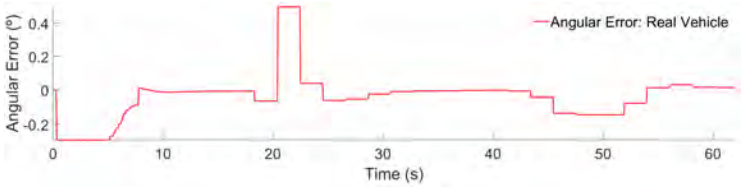


(d) Platoon B: FSM states of each vehicle.

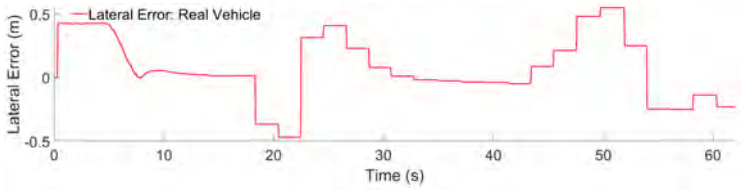
Fig. 5.30: Longitudinal performance of Platoon B during the platoon lane merging scenario, using the RTTP approach combined with the car following strategy in a mixed test environment.

100 % in detecting DDoS attacks, evidencing its capability to detect and report such threats in real-time. The Secure Gateway similarly exhibited effective detection and alerting capacities.

Moreover, the modules installed in the CS showed their efficacy. The honeypot system was highly successful in identifying security vulnerabilities, with a detection rate of 100 % for both port scanning and login attack attempts. The FEAM module reported an unauthorized device access rate of 0 %, which is the desired outcome, as the CS should only be accessible to authorized users due to its ability to alter road parameters and control platoon formation. Lastly, the RMS successfully logged the data recorded during the tests. Although no issues or malfunctions were detected during the tests, the RMS correctly reported threats when data was deliberately modified to test the module's functionality.



(a) Merging Vehicle: angular error of the lane change maneuver.



(b) Merging Vehicle: lateral error of the lane change maneuver

Fig. 5.31: Lateral performance of the merging vehicle during the platoon lane merging, using the RTTP approach alongside the car following strategy in a mixed test environment.

5.3.4 Discussions

This section evaluated the performance of the RTTP method in executing a platoon lane merging maneuver within both simulation and mixed test environments. To better comprehend the results obtained, Table 5.4 is presented, where different metrics are studied in order to evaluate the overall performance of the merging vehicles in each tests. These vehicles are the most interesting, since they executes the most amount of operations during the maneuver. Furthermore, the table adds the results obtained from this maneuver using the HYTP method, to obtain a better perspective of the RTTP performance. It is worth noting that metrics evaluated corresponds to the period before the platoon lane merging starts until they vehicles are stable in the newly formed platoon.

When compared to the HYTP method, the RTTP demonstrated superior performance in terms of execution time and lateral displacement. The RTTP executed the lane merging in 3 s and completed the entire maneuver in 21 s, compared to the HYTP, which took 12 s for lane merging and 25 s for the complete maneuver. This reduced time is attributed to the RTTP's mid-term trajectory generated with a 3 s horizon view for lane changes. Additionally, the RTTP adjusts the position in parallel, whereas in the HYTP, Platoon A first opens the gap, followed by Platoon B adjusting its position. The mean angular and lateral error of each vehicle with the RTTP were less than 0.01^o and 0.01 m respectively, compared to the 0.682^o and 0.241 m of the HYTP for the simulation. This is mainly due to the HYTP's MPC adjusting the references for executing lane changes, thus producing more errors, compared to the RTTP's intermediate trajectory generated for moving from one lane to another.

Both methods demonstrated effective performance in car following maneuvers, maintaining a non-amplified speed change even during cut-in and cut-out maneuvers that opened and closed gaps. However, the RTTP method outperformed the HYTP

method, as each vehicle presented lower errors, particularly with respect to the mean distance and time gap errors.

Tab. 5.4: Mean error comparative of the merging vehicles during the platoon lane merging maneuver corresponding to each test.

	Simulation Test with HYTP		**Simulation Test with RTTP**			**Mix Test Environment Case 1**	**Mix Test Environment Case 2**
Metric	Vehicle 2	Vehicle 4	Vehicle 2	Vehicle 4	Vehicle 6	Twizy	Twizy
Angular Error (º)	0.6827	0.682	0.0028	0.0027	0.0027	0.0497	0.0765
Lateral Error (m)	0.2415	0.2415	0.0066	0.0062	0.0058	0.14	0.1965
Distance Error (m)	1.6993	3.3412	0.0968	0.492	1.002	0.9309	0.859
Speed Error (m/s)	0.131	0.3408	0.1815	0.3192	0.6437	0.2765	0.4235
Time Gap Error (s)	0.4282	0.8696	0.0206	0.1157	0.255	0.5394	0.3265
Lane Change Duration (s)	12	12	3	3	3	3	3
Maneuver Duration (s)	25	25	21	21	21	10	6

In the context of mixed-environment tests, the conducted maneuvers are generally successful. The real vehicle effectively accomplishes the merging process in each position while maintaining a safe inter-vehicle distance within the platoon. Furthermore, the real vehicle completes the lane change within the anticipated 3 s time frame, exhibiting minimal lateral and angular errors. The total maneuver times are 10 s when the vehicle merges in the middle and 6 s when the vehicle merges from behind. These differences can be attributed to the necessity of adjustments from Platoon A in one scenario, which is not required in the other.

The performance errors of the car following controller were relatively low with regards to speed and distance, being less than 0.5 m/s and 1 m respectively. However, the time gap exhibited a slightly higher error margin of 0.5394 s and 0.3265 s, indicating more difficulty in tracking this parameter. It is important to note that an issue was encountered with the model used in CARLA, which exhibited instability in

low-speed scenarios. This issue could potentially be addressed through enhancements to the data concerning the dynamics of the Twizy, thereby improving its behavior and, consequently, the overall maneuver performance. Despite this challenge, the maneuver was successfully completed during the test.

Upon thorough examination, performance metrics obtained prior to and subsequent to the incorporation of the IoTAC modules exhibited no significant variations. This substantiates that the implementation of additional security provisions facilitated by the IoTAC platform did not adversely impact the efficiency or efficacy of the platooning maneuvers, thereby affirming the robustness and adaptability of the driving architecture and the IoTAC system. Moreover, the individual modules demonstrated notable performance, accurately identifying and signaling potential threats.

In summary, the RTTP method demonstrates substantial enhancements in both its range of applications and performance when compared to the HYTP method. The RTTP method exhibits minimal errors and is capable of executing maneuvers with considerable speed. Nonetheless, upon transitioning to real vehicles, there are opportunities for further refinement, particularly in terms of car following control in the context of tight curves.

5.4 Summary and Conclusions

This chapter presents the experimental findings of the Car Following control algorithm and two decision methodologies for cooperative maneuvers, namely, the HYTP and RTTP. These methodologies were evaluated across diverse testing platforms, encompassing simulated, real, and hybrid environments. It should be noted that these developments were also employed in an variety of European and national projects.

The car following control algorithm, incorporating ACC and CACC technologies, showed string stable performance. Trials with two Twizys confirmed the control effectiveness in real world applications, maintaining speed, distance, and time gap. The algorithm also functioned safely with a 12-m bus, despite errors, suggesting the need for more data and diverse vehicle models to expand the algorithm's capabilities to heterogeneous platoons. Finally, an urban simulation demonstrated the effects of the negotiation process, with vehicles transitioning from automated mode (*lane keeping* state) to ACC and then CACC mode (*car following state*).

The research evaluates two decision methods, focusing primarily on merging maneuvers. The first method, HYTP, underwent testing in simulation and mixed environments. Despite the real vehicle overshooting acceleration MPC bounds, it safely executed the roundabout entry within comfort driving standards, prompting the need for new MPC configuration parameters. In case of the platoon lane merging, The Platoon B successfully merged into the main lane, demonstrating the

decentralized method's low computational cost. However, improvements are needed in lane change execution time and vehicle positioning post-change.

Integrating the RTTP with the existing car following algorithm enhanced the HYTP process's performance. Simulation tests showed superior capabilities, notably reducing lane change duration and total execution time compared to the HYTP method. Both methods exhibited good car following capabilities. Testing in a mixed environment yielded positive results, with the real vehicle executing the merging within less than 10 s and the lane change within 3 s, despite encountering instability with the CARLA model in low-speed scenarios. This issue could be addressed by refining Twizy dynamics and behavior data, but overall, the maneuver was successfully achieved.

The investigation into the SerIoT and IoTAC systems has underscored the adaptability and efficacy of the proposed driving architecture. Its versatility allows it to effectively meet the demands of different cybersecurity systems, thus facilitating the demonstration of their capabilities. The architecture has demonstrated its proficiency, regardless of the decision method employed, in safely executing a variety of maneuvers.

The outcomes of this research are both promising and encouraging. The successful execution of maneuvers across various testing environments and platforms substantiates the effectiveness of the decision and control algorithm in managing a diverse array of driving situations. Crucially, the accomplishments achieved in platoon lane merging, facilitated by the Hybrid Trajectory Approach, culminated in a published paper [124]. Additionally, the findings pertaining to the SerIoT system also led to another publication [188].

Conclusions

6

"If you feel hitting up against your limit, remember for what cause you clench your fist! Remember why you started down this path, and let that memory carry you beyond your limit."
- All Might

This chapter summarizes the main contributions of this Ph.D. Thesis, based on the review of the SoA in the driving architecture of CCAM, as well as cooperative maneuvers presented in Chapter 2, the validation framework to test cooperative maneuver presented in Chapter 3, the cooperative maneuver decision and control developments described in Chapter 4, and finally, the validation tests results in the different test environments presented in Chapter 5. Furthermore, recommendations and future works derived from this thesis are delivered.

6.1 Concluding Remarks

CCAM technologies were developed to revolutionize transportation by integrating communication and automation, enhancing safety, efficiency, and sustainability. They address challenges like traffic congestion and accidents while optimizing infrastructure use. CCAM aims to provide seamless mobility experiences and improve overall service quality. By fostering collaboration among vehicles and infrastructure, CCAM creates adaptable transportation networks.

In order to achieve these objectives, various architectural definitions that compose these systems have been proposed over time, with primary areas of study being the perception, decision, and control modules. Nevertheless, the increasing relevance of cooperation between vehicles and infrastructure in resolving complex scenarios has led to further exploration of both the communication and infrastructure modules, thereby introducing new challenges.

This thesis reviews each component that compose the architecture, as well the cooperative aspect of CCAM, examining various maneuvers such as car following, intersection management, cooperative overtaking, and cooperative merging. It identifies car following and cooperative merging as two areas of significant interest. Car following currently holds great relevance due to its potential in optimizing traffic and enhancing safety, thus making maneuvers involving vehicle platoons an interesting topic. On the other hand, cooperative merging focuses more on achieving

safe and optimal conduct on certain types of road segments, making it an essential maneuver for optimal driving. Several challenges, including cyber-security threats, regulatory compliance concerns, public acceptance issues, infrastructural challenges, the coverage of current algorithms, and the lack of real-world tests were identified within these maneuvers.

This thesis is focused on the development of algorithms for cooperative maneuvers, specifically those pertaining to car following and cooperative merging. The application of these maneuvers in various testing scenarios, including real-world, simulated, and mixed test environments, is thoroughly investigated. Moreover, this research explores cyber-security measures associated with these maneuvers. To this end, the AUDRIC architecture is proposed as the framework to tackle these problems. Notably, at the beginning of this thesis, the AUDRIC architecture was lacking of cooperative capabilities and infrastructure integration. Therefore, the efforts were directed towards enhancing these qualities within the architecture, which also involved the adaptation of the architecture to accommodate cyber-secure systems such as SerIoT and IoTAC. One of AUDRIC's key traits is its compatibility with diverse platforms, both simulated and real, which were utilized to validate the thesis's developments. In terms of simulation, ROS + CARLA emerged as the preferred option due to their superior ability to simulate a wide variety of complex scenarios compared to MATLAB/Simulink + Dynacar, which was initially used by the CCAM group.

This thesis presents a new car following strategy. A feedforward/feedback + PD controller was selected for its robust performance and adaptability to various decision strategies. Subsequently, two decision methods are presented. The first, HYTP, was the main decision strategy within the CCAM group at the beginning of this work. It incorporates nominal static trajectories based on Bézier Curves, which are adjusted by a predictive maneuver planner. This planner was adapted in this thesis to execute maneuvers such as merging. The second method, RTTP, uses a detailed map and an A* search algorithm to generate a drivable space. This space is then modified by a behavioral planner, which incorporates a FSM with states representing maneuvers, namely *lane keeping*, *car following*, and *merging*. A local planner, employing real-time trajectories based on Bézier curves, is also integrated. RTTP allows for the execution of complex scenarios and adheres more closely to navigational processes presented in existing literature than the HYTP. After describing both decision methodologies, the logic guiding the negotiation of maneuvers is introduced, which includes the use of V2X messages, specifically the PMM and PCM. These messages were adapted for urban scenarios with different vehicle types.

Finally, the validation tests for the strategies developed for cooperative maneuvers demonstrated a shared relationship with various EU and national projects, validating the developments within these projects while simultaneously being guided by their

objectives. Initially, the performance of the car following strategy was evaluated in both simulated and real scenarios, followed by the testing of the complete strategy involving the RTTP-FSM decision approach. This approach demonstrated overall strong performance and adaptability to different platforms and scenarios. Subsequently, the HYTP method was tested in a variety of scenarios, such as roundabout merging, platoon lane merging, fleet management, and intersection management. The HYTP method successfully executed each maneuver, demonstrating safety and comfort, although there remains room for improvement. This led to the final decision method, the RTTP, which showcased superior performance in platoon lane merging maneuvers in simulated and mixed test environments, surpassing the HYTP in each evaluated metric. Ultimately, the investigation into the SerIoT and IoTAC system highlighted the adaptability of the complete driving architecture, while proving the benefits of these systems in the CCAM field.

Upon concluding the final tests related to the RTTP, this thesis has successfully met its primary objective and each of the secondary objectives outlined during its beginning. Furthermore, throughout this process, each test performed has contributed to fulfilling the goals of both European and national projects involved. However, research is an ongoing process and, therefore, potential future studies building on the foundation of this thesis are proposed.

6.2 Research Perspective and Future Works

This thesis has comprehensively addressed the area of cooperative maneuvers in a diverse amount of testing cases, while briefly addressing aspects pertaining to cyber-security. In this sense, researchers can use this work as a reference for future developments to extend the robustness and applicability of these studies. In this context, the proposed future works are:

- **Integrate more cooperative maneuvers**: The current study primarily focuses on car following and merging scenarios, which cover a substantial range of driving situations. However, to fully encompass the diverse driving scenarios encountered in real-world, additional cooperative maneuvers must be addressed. One such maneuver is intersection management, which can be approached from both a vehicular and infrastructural perspective. For example, when considering a platoon of vehicles, the leader must determine the optimal speed to ensure the entire platoon successfully navigates the intersection. In the case of a single vehicle, the process may vary. Another critical maneuver to consider is overtaking, particularly in scenarios involving platoons. This represents a relatively under-researched area and could potentially optimize traffic flow.
- **Improves the car following strategy**: The car following controller selected, while robust and widely used, may not always yield optimal performance. Its effectiveness heavily relies on the specific vehicle model, which could pose a

drawback when applying the strategy to other scenarios. Tuning strategies, such as the one used in [104], can alleviate this issue to some extent. However, a controller that isn't dependent on the vehicle model may be desirable for its ability to be extrapolated to different scenarios. Crucially, such a strategy should be able to maintain the string stability criteria.

- **Improves and expand the merging strategy**: While the strategies for roundabout merging and lane merging have been tested, there are other merging scenarios that warrant further exploration. A key example is the merging of platoons in a roundabout, a promising approach that could potentially mitigate conflicts at roundabout entrances. Another area of interest is on-ramp merging, both for single vehicles and platoons. Although existing literature has examined on-ramp merging scenarios, as of the time of this writing, no successful algorithm has been developed that adequately addresses all three merging situations.
- **Expand the Decision strategy**: The RTTP approach demonstrates a robust performance to handle diverse scenarios, encompassing both cooperative and non-cooperative contexts. However, the standalone FSM employed as a decision-making strategy may exhibit limitations when additional maneuvers are introduced or when determining the optimal decision. An emerging solution involves the integration of a FSM with AI. Despite AI algorithms possessing their own limitations, the combination of these two strategies may prove beneficial in handling complex cases involving multiple vehicles.
- **Enhancing the testing methodologies**: The model used in conjunction with CARLA exhibits instability at slow speeds, which may negatively impact algorithm validation. A suggested enhancement would involve collecting more real-world data from the platform to refine the model, thereby providing a more accurate representation. Additionally, the capabilities of CARLA in simulating vehicles, objects, pedestrians, etc. could be used to perform more complex simulations before implementing the algorithms in real-world scenarios. Furthermore, examining more vehicle types could help to broaden the interoperability of the architecture.
- **Explore more cyber-security frameworks**: While internet connectivity offers significant advantages, it also introduces risks that must be addressed. Both cyber-security frameworks explored in this thesis demonstrated certain limitations, particularly in the realm of detection and mitigation times. Given the critical nature of CCAM, where milliseconds can be consequential, these frameworks' response times were found to be sub-optimal. Therefore, with ongoing technological advancements, the exploration of new cyber-security frameworks is essential to ensure comprehensive protection for CCAM agents.

Bibliography

[1] Sofia Gaitanidou and George Yannis. “Traffic Congestion and Road Accidents in Europe”. In: *Transportation Research Procedia* 55 (2021), pp. 82–89.

[2] Mohamed Abdel-Aty, Jaeyoung Lee, and Nasim Uddin. “Analyzing the relationship between traffic congestion and crashes in urban areas”. In: *Accident Analysis & Prevention* 153 (2021), p. 105968.

[3] Bob Pishue. “2022 INRIX global traffic scorecard”. In: *INRIX (January 2023)*. 2023.

[4] Li Li, Zai Zhang, Zhi-Gang Xu, Wen-Chen Yang, and Qing-Chang Lu. “The role of traffic conflicts in roundabout safety evaluation: a review”. In: *Accident Analysis & Prevention* 196 (2024), p. 107430.

[5] G Brusaglino. “Safe and effective mobility in Europe-the contribution of the PROMETHEUS programme”. In: *IEE Colloquium on Prometheus and Drive*. IET. 1992, pp. 1–1.

[6] Ming Xie, Laurent Trassoudaine, Joseph Alizon, Monique Thonnat, and Jean Gallice. “Active and intelligent sensing of road obstacles: Application to the European Eureka-PROMETHEUS project”. In: *1993 (4th) International Conference on Computer Vision*. IEEE. 1993, pp. 616–623.

[7] Jan P van Dijke and Margriet van Schijndel. “Citymobil, advanced transport for the urban environment: Update”. In: *Transportation research record* 2324.1 (2012), pp. 29–36.

[8] Adriano Alessandrini, Alessio Cattivera, Carlos Holguin, and Daniele Stam. “CityMobil2: challenges and opportunities of fully automated mobility”. In: *Road vehicle automation* (2014), pp. 169–184.

[9] Christian Rösener, Adrian Zlocki, Hendrik Weber, and Johannes Hiller. “Evaluation of Automated Driving by Large-Scale Piloting on European Roads–The L3Pilot Project”. In: *Road Vehicle Automation 6 6*. Springer. 2019, pp. 75–83.

[10] Martin Skoglund, Anders Thorsen, Alvaro Arrue, et al. “Technical and functional requirements for V2X communication, positioning and cyber-security in the HEADSTART project”. In: *27th ITS World Congress, Hamburg, Germany, 11-15 October 2021*. 2021.

[11] A Gräter, M Harrer, M Rosenquist, and E Steiger. “Connected, Cooperative and Automated Mobility Roadmap”. In: *European Road: Transport Reserach Advisory Council* (2022).

[12] Antonius JC Schmeitz, Dehlia MC Willemsen, and Simon Ellwanger. “EU ENSEMBLE Project: Reference Design and Implementation of the Platooning Support Function”. In: *IEEE Transactions on Intelligent Transportation Systems* (2023).

[13] A. Anund, R. Ludovic, B. Caroleo, et al. “Lessons learned from setting up a demonstration site with autonomous shuttle operation – based on experience from three cities in Europe”. In: *Journal of Urban Mobility*. Vol. 2. ELSEVIER. 2022.

[14] Erol Gelenbe, Joanna Domanska, Tadeusz Czàchorski, Anastasis Drosou, and Dimitrios Tzovaras. “Security for internet of things: The seriot project”. In: *2018 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE. 2018, pp. 1–5.

[15] Miltiadis Siavvas, Erol Gelenbe, Dimitrios Tsoukalas, et al. “The IoTAC software security-by-design platform: Concept, challenges, and preliminary overview”. In: *2022 18th International Conference on the Design of Reliable Communication Networks (DRCN)*. IEEE. 2022, pp. 1–6.

[16] Lingyun Xiao and Feng Gao. “A comprehensive review of the development of adaptive cruise control systems”. In: *Vehicle system dynamics* 48.10 (2010), pp. 1167–1192.

[17] B. Smith et al. “A brief history of self-driving cars”. In: *Nature* 573.7772 (2019), pp. 371–375.

[18] SAE On-Road Automated Vehicle Standards Committee. *J3016: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. Tech. rep. J3016-202104. SAE International, 2021, pp. 01–16.

[19] David González Bautista, Joshué Pérez, Vicente Milanés, and Fawzi Nashashibi. “A Review of Motion Planning Techniques for Automated Vehicles”. In: *IEEE Transactions on Intelligent Transportation Systems* (2016).

[20] Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, et al. “Stanley: The robot that won the DARPA Grand Challenge”. In: *Journal of field Robotics* 23.9 (2006), pp. 661–692.

[21] José Ángel Matute Peaspán. “Design and validation of decision and control systems in automated driving”. PhD thesis. Universidad del País Vasco-Euskal Herriko Unibertsitatea, 2021.

[22] Bernhard Häfner, Julian Sauerhammer, Georg A Schmitt, and Jörg Ott. “Performance of cooperative maneuver protocols in real-world automated vehicles”. In: *ICC 2022-IEEE International Conference on Communications*. IEEE. 2022, pp. 1494–1499.

[23] Zhengwei Bai, Peng Hao, Wei Shangguan, Baigen Cai, and Matthew J Barth. “Hybrid reinforcement learning-based eco-driving strategy for connected and automated vehicles at signalized intersections”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.9 (2022), pp. 15850–15863.

[24] Nicolas Scheiner, Florian Kraus, Nils Appenrodt, Jürgen Dickmann, and Bernhard Sick. “Object detection for automotive radar point clouds–a comparison”. In: *AI Perspectives* 3.1 (2021), pp. 1–23.

[25] Daniel Bogdoll, Maximilian Nitsche, and J Marius Zöllner. “Anomaly detection in autonomous driving: A survey”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 4488–4499.

[26] Chaoyang Wang, Xiaonan Wang, Hao Hu, Yanxue Liang, and Gang Shen. “On the Application of Cameras Used in Autonomous Vehicles”. In: *Archives of Computational Methods in Engineering* 29.6 (2022), pp. 4319–4339.

[27] Xun Yang, Yunyang Shi, Jiping Xing, and Zhiyuan Liu. “Autonomous driving under V2X environment: state-of-the-art survey and challenges”. In: *Intelligent Transportation Infrastructure* 1 (2022).

[28] Piotr Szymanski, Biagio Ciuffo, Georgios Fontaras, Giorgio Martini, and Ferenc Pekar. “The future of road transport in Europe. Environmental implications of automated, connected and low-carbon mobility”. In: *Combustion Engines* 60 (2021).

[29] Liang Chen, Fu Zheng, Xiaopeng Gong, and Xinyuan Jiang. “GNSS High-Precision Augmentation for Autonomous Vehicles: Requirements, Solution, and Technical Challenges”. In: *Remote Sensing* 15.6 (2023), p. 1623.

[30] Shuran Zheng, Jinling Wang, Chris Rizos, Weidong Ding, and Ahmed El-Mowafy. “Simultaneous Localization and Mapping (SLAM) for Autonomous Driving: Concept and Analysis”. In: *Remote Sensing* 15.4 (2023), p. 1156.

[31] Guotao Xie, Jing Zhang, Junfeng Tang, et al. “Obstacle detection based on depth fusion of lidar and radar in challenging conditions”. In: *Industrial Robot: the international journal of robotics research and application* 48.6 (2021), pp. 792–802.

[32] Ricardo Roriz, Jorge Cabral, and Tiago Gomes. “Automotive LiDAR technology: A survey”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.7 (2021), pp. 6282–6297.

[33] Sampo Kuutti, Saber Fallah, Konstantinos Katsaros, et al. “A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications”. In: *IEEE Internet of Things Journal* 5.2 (2018), pp. 829–846.

[34] Anaïs Halin, Jacques G Verly, and Marc Van Droogenbroeck. “Survey and synthesis of state of the art in driver monitoring”. In: *Sensors* 21.16 (2021), p. 5558.

[35] Pranav Kumar Singh, Sunit Kumar Nandi, and Sukumar Nandi. “A tutorial survey on vehicular communication state of the art, and future research directions”. In: *Vehicular Communications* 18 (2019), p. 100164.

[36] Felipe Domingos Da Cunha, Azzedine Boukerche, Leandro Villas, Aline Carneiro Viana, and Antonio AF Loureiro. “Data communication in VANETs: a survey, challenges and applications”. In: (2014).

[37] Christian Cseh. “Architecture of the dedicated short-range communications (DSRC) protocol”. In: *VTC’98. 48th IEEE Vehicular Technology Conference. Pathway to Global Wireless Revolution (Cat. No. 98CH36151)*. Vol. 3. IEEE. 1998, pp. 2095–2099.

[38] “IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”. In: *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)* (2012), pp. 1–2793.

[39] SM Suhail Hussain, Taha Selim Ustun, Paul Nsonga, and Ikbal Ali. “IEEE 1609 WAVE and IEC 61850 standard communication based integrated EV charging management in smart grids”. In: *IEEE Transactions on Vehicular Technology* 67.8 (2018), pp. 7690–7697.

[40] Ali J Ghandour, Marco Di Felice, Hassan Artail, and Luciano Bononi. “Dissemination of safety messages in IEEE 802.11 p/WAVE vehicular network: Analytical study and protocol enhancements”. In: *Pervasive and mobile computing* 11 (2014), pp. 3–18.

[41] SAE On-Road Automated Vehicle Standards Committee. *J2735: V2X Communications Message Set Dictionary*. Tech. rep. J2735-202309. SAE International, 2023, pp. 12–83.

[42] Suryansh Saxena and Isaac K Isukapati. “Simulated Basic Safety Message: Concept & Application”. In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2019, pp. 2450–2456.

[43] Sebastian Kuhlmorgen, Ignacio Llatser, Andreas Festag, and Gerhard Fettweis. “Performance evaluation of etsi geonetworking for vehicular ad hoc networks”. In: *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*. IEEE. 2015, pp. 1–6.

[44] José Santa, Fernando Pereñíguez, Antonio Moragón, and Antonio F Skarmeta. “Experimental evaluation of CAM and DENM messaging services in vehicular communications”. In: *Transportation Research Part C: Emerging Technologies* 46 (2014), pp. 98–120.

[45] Tamás Wágner, Tamás Ormándi, Tamás Tettamanti, and István Varga. “SPaT/MAP V2X communication between traffic light and vehicles and a realization with digital twin”. In: *Computers and Electrical Engineering* 106 (2023), p. 108560.

[46] Oliver Brandl. “V2X traffic management”. In: *e & i Elektrotechnik und Informationstechnik* 133.7 (2016), pp. 353–355.

[47] José Roldán-Gómez, Javier Carrillo-Mondéjar, Juan Manuel Castelo Gómez, and Sergio Ruiz-Villafranca. “Security Analysis of the MQTT-SN Protocol for the Internet of Things”. In: *Applied Sciences* 12.21 (2022), p. 10991.

[48] Yuankai He, Baofu Wu, Zheng Dong, Jian Wan, and Weisong Shi. “Towards C-V2X Enabled Collaborative Autonomous Driving”. In: *IEEE Transactions on Vehicular Technology* (2023).

[49] Valerian Mannoni, Vincent Berg, Stefania Sesia, and Eric Perraud. “A comparison of the V2X communication systems: ITS-G5 and C-V2X”. In: *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. IEEE. 2019, pp. 1–5.

[50] Johann M Marquez-Barja, Dries Naudts, Vasilis Maglogiannis, et al. “Designing a 5G architecture to overcome the challenges of the teleoperated transport and logistics”. In: *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE. 2022, pp. 264–267.

[51] Matti Kutila, Pasi Pyykonen, Qing Huang, et al. “C-V2X supported automated driving”. In: *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2019, pp. 1–5.

[52] Saqib Hakak, Thippa Reddy Gadekallu, Praveen Kumar Reddy Maddikunta, et al. “Autonomous Vehicles in 5G and beyond: A Survey”. In: *Vehicular Communications* (2022), p. 100551.

[53] Grigorios Kakkavas, Kwame Nseboah Nyarko, Charbel Lahoud, et al. “Teleoperated support for remote driving over 5G mobile communications”. In: *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. IEEE. 2022, pp. 280–285.

[54] Dirk Hetzer, Maciej Muehleisen, Apostolos Kousaridas, et al. “5G connected and automated driving: use cases, technologies and trials in cross-border environments”. In: *EURASIP Journal on Wireless Communications and Networking* 2021.1 (2021), pp. 1–19.

[55] Benedikt Brecht, Dean Therriault, André Weimerskirch, et al. “A security credential management system for V2X communications”. In: *IEEE Transactions on Intelligent Transportation Systems* 19.12 (2018), pp. 3850–3871.

[56] Lei Chen and Cristofer Englund. “Cooperative ITS—EU standards to accelerate cooperative mobility”. In: *2014 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE. 2014, pp. 681–686.

[57] Takahito Yoshizawa, Dave Singelée, Jan Tobias Muehlberg, et al. “A survey of security and privacy issues in v2x communication systems”. In: *ACM Computing Surveys* 55.9 (2023), pp. 1–36.

[58] Zeinab El-Rewini, Karthikeyan Sadatsharan, Daisy Flora Selvaraj, Siby Jose Plathottam, and Prakash Ranganathan. “Cybersecurity challenges in vehicular communications”. In: *Vehicular Communications* 23 (2020), p. 100214.

[59] Ray Alejandro Lattarulo Arias. “Validation of trajectory planning strategies for automated driving under cooperative, urban, and interurban scenarios.” In: (2019).

[60] Fernando Garrido and Paulo Resende. “Review of Decision-Making and Planning Approaches in Automated Driving”. In: *IEEE Access* 10 (2022), pp. 100348–100366.

[61] Grigorios D Konstantakopoulos, Sotiris P Gayialis, and Evripidis P Kechagias. “Vehicle routing problem and related algorithms for logistics distribution: A literature review and classification”. In: *Operational research* (2020), pp. 1–30.

[62] Hannah Bast, Daniel Delling, Andrew Goldberg, et al. “Route planning in transportation networks”. In: *Algorithm engineering: Selected results and surveys* (2016), pp. 19–80.

[63] Gilbert Laporte and Yves Nobert. “Exact algorithms for the vehicle routing problem”. In: *North-Holland mathematics studies*. Vol. 132. Elsevier, 1987, pp. 147–184.

[64] Shahab Karimi and Ardalan Vahidi. “Monte Carlo Tree Search and Cognitive Hierarchy Theory for Interactive-Behavior Prediction in Fast Trajectory Planning and Automated Lane Change”. In: *Journal of Autonomous Vehicles and Systems* 1.1 (2021), p. 011008.

[65] Zheng Zhang, Juan Chen, and Qing Guo. “Agvs route planning based on region-segmentation dynamic programming in smart road network systems”. In: *Scientific Programming* 2021 (2021), pp. 1–13.

[66] PAN Ping-Qi. *Linear programming computation*. Springer, 2014.

[67] Seyed Alireza Fayazi and Ardalan Vahidi. “Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing”. In: *IEEE Transactions on Intelligent Vehicles* 3.3 (2018), pp. 287–299.

[68] Ivan Kantor, Jean-Loup Robineau, Hür Bütün, and Francois Marechal. “A mixed-integer linear programming formulation for optimizing multi-scale material and energy integration”. In: *Frontiers in Energy Research* 8 (2020), p. 49.

[69] Fei Liu, Chengyu Lu, Lin Gui, et al. “Heuristics for Vehicle Routing Problem: A Survey and Recent Advances”. In: *arXiv preprint arXiv:2303.04147* (2023).

[70] Tomoya Kawabe, Tatsushi Nishi, and Ziang Liu. “Flexible Route Planning for Multiple Mobile Robots by Combining Q–Learning and Graph Search Algorithm”. In: *Applied Sciences* 13.3 (2023), p. 1879.

[71] Meng-Yue Zhang, Shi-Chun Yang, Xin-Jie Feng, et al. “Route Planning for Autonomous Driving Based on Traffic Information via Multi-Objective Optimization”. In: *Applied Sciences* 12.22 (2022), p. 11817.

[72] Diange Yang, Xinyu Jiao, Kun Jiang, and Zhong Cao. “Driving space for autonomous vehicles”. In: *Automotive Innovation* 2 (2019), pp. 241–253.

[73] Yanjun Huang, Jiatong Du, Ziru Yang, et al. “A survey on trajectory-prediction methods for autonomous driving”. In: *IEEE Transactions on Intelligent Vehicles* 7.3 (2022), pp. 652–674.

[74] Bernhard Häfner, Vaibhav Bajpai, Jörg Ott, and Georg A Schmitt. “A survey on cooperative architectures and maneuvers for connected and automated vehicles”. In: *IEEE Communications Surveys & Tutorials* 24.1 (2021), pp. 380–403.

[75] David González, Joshué Pérez, Vicente Milanés, and Fawzi Nashashibi. “A review of motion planning techniques for automated vehicles”. In: *IEEE Transactions on intelligent transportation systems* 17.4 (2015), pp. 1135–1145.

[76] Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, et al. “Self-driving cars: A survey”. In: *Expert Systems with Applications* 165 (2021), p. 113816.

[77] Wei Liu, Min Hua, Zhiyun Deng, et al. “A systematic survey of control techniques and applications: From autonomous vehicles to connected and automated vehicles”. In: *arXiv preprint arXiv:2303.05665* (2023).

[78] Eugene Lavretsky, Kevin A Wise, Eugene Lavretsky, and Kevin A Wise. “Optimal control and the linear quadratic regulator”. In: *Robust and Adaptive Control: With Aerospace Applications* (2013), pp. 27–50.

[79] Weida Wang, Taiheng Ma, Chao Yang, et al. “A path following lateral control scheme for four-wheel independent drive autonomous vehicle using sliding mode prediction control”. In: *IEEE Transactions on Transportation Electrification* 8.3 (2022), pp. 3192–3207.

[80] Kai Yang, Xiaolin Tang, Yechen Qin, et al. “Comparative study of trajectory tracking control for automated vehicles via model predictive control and robust H-infinity state feedback control”. In: *Chinese Journal of Mechanical Engineering* 34.1 (2021), pp. 1–14.

[81] Gonçalo Collares Pereira, Bo Wahlberg, Henrik Pettersson, and Jonas Mårtensson. "Adaptive reference aware MPC for lateral control of autonomous vehicles". In: *Control Engineering Practice* 132 (2023), p. 105403.

[82] András Ratkovics, Sándor Vass, and Viktor Tihanyi. "Development of an LQ regulator for longitudinal vehcile control of an automated vehicle". In: *2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo)*. IEEE. 2019, pp. 000149–000154.

[83] Mohammad Rokonuzzaman, Navid Mohajer, Shady Mohamed, and Saeid Nahavandi. "A customisable longitudinal controller of autonomous vehicle using data-driven mpc". In: *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2021, pp. 1367–1373.

[84] Martin Buechel and Alois Knoll. "Deep reinforcement learning for predictive longitudinal control of automated vehicles". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2018, pp. 2391–2397.

[85] Jose A Matute, Ray Lattarulo, Asier Zubizarreta, and Joshue Perez. "A comparison between coupled and decoupled vehicle motion controllers based on prediction models". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2019, pp. 1843–1848.

[86] Ardashir Mohammadzadeh and Hamid Taghavifar. "A robust fuzzy control approach for path-following control of autonomous vehicles". In: *Soft Computing* 24.5 (2020), pp. 3223–3235.

[87] Zhaoxuan Zhu, Shobhit Gupta, Abhishek Gupta, and Marcello Canova. "A deep reinforcement learning framework for eco-driving in connected and automated hybrid electric vehicles". In: *IEEE Transactions on Vehicular Technology* (2023).

[88] Arvind Goyal and Ayushi Thakur. "An overview of drive by wire technology for automobiles". In: *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*. IEEE. 2019, pp. 108–110.

[89] Jose Angel Matute-Peaspan, Asier Zubizarreta-Pico, and Sergio E Diaz-Briceno. "A vehicle simulation model and automated driving features validation for low-speed high automation applications". In: *IEEE Transactions on Intelligent Transportation Systems* 22.12 (2020), pp. 7772–7781.

[90] Klaus Bengler, Michael Rettenmaier, Nicole Fritz, and Alexander Feierle. "From HMI to HMIs: Towards an HMI framework for automated driving". In: *Information* 11.2 (2020), p. 61.

[91] Karthik Mahadevan, Sowmya Somanath, and Ehud Sharlin. "Communicating awareness and intent in autonomous vehicle-pedestrian interaction". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 2018, pp. 1–12.

[92] Michael Rettenmaier, Moritz Pietsch, Jonas Schmidtler, and Klaus Bengler. "Passing through the bottleneck-the potential of external human-machine interfaces". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2019, pp. 1687–1692.

[93] Tomislav Mihalj, Hexuan Li, Dario Babić, et al. "Road Infrastructure Challenges Faced by Automated Driving: A Review". In: *Applied Sciences* 12.7 (2022), p. 3477.

[94] Hieu Ngo, Hua Fang, and Honggang Wang. "Cooperative Perception With V2V Communication for Autonomous Vehicles". In: *IEEE Transactions on Vehicular Technology* (2023).

[95] Wenbo Chu, Qiqige Wuniri, Xiaoping Du, et al. "Cloud control system architectures, technologies and applications on intelligent and connected vehicles: a review". In: *Chinese Journal of Mechanical Engineering* 34.1 (2021), pp. 1–23.

[96] Domagoj Majstorović, Simon Hoffmann, Florian Pfab, et al. “Survey on teleoperation concepts for automated vehicles”. In: *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2022, pp. 1290–1296.

[97] Seyed Mehdi Meshkani and Bilal Farooq. “Centralized and decentralized algorithms for two-to-one matching problem in ridehailing systems”. In: *EURO Journal on Transportation and Logistics* 12 (2023), p. 100106.

[98] Daniel Maksimovski, Christian Facchi, and Andreas Festag. “Cooperative Driving: Research on Generic Decentralized Maneuver Coordination for Connected and Automated Vehicles”. In: *International Conference on Vehicle Technology and Intelligent Transport Systems*. Springer. 2021, pp. 348–370.

[99] Nils Dreyer, Andreas Moller, Zeeshan Hameed Mir, Fethi Filali, and Thomas Kurner. “A data traffic steering algorithm for IEEE 802.11 p/LTE hybrid vehicular networks”. In: *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. IEEE. 2016, pp. 1–6.

[100] Masaya Mizutani, Manabu Tsukada, and Hiroshi Esaki. “Automcm: Maneuver coordination service with abstracted functions for autonomous driving”. In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 1069–1076.

[101] Rafael Molina-Masegosa, Sergei S Avedisov, Miguel Sepulcre, et al. “V2X Communications for Maneuver Coordination in Connected Automated Driving: Message Generation Rules”. In: *IEEE Vehicular Technology Magazine* (2023).

[102] Dehlia MC Willemsen, Antonius JC Schmeitz, and Edoardo Mascalchi. “EU ENSEMBLE Project: Specification of an Interoperable Solution for a Support Function for Platooning”. In: *IEEE Transactions on Intelligent Transportation Systems* (2023).

[103] International Organization for Standardization. *ISO 4272: Intelligent transport systems — Truck platooning systems (TPS) — Functional and operational requirements*. Tech. rep. ISO4272-2022. ISO, 2022, pp. –.

[104] Carlos Flores. “Architecture de contrôle pour le car-following adaptatif et coopératif Control architecture for adaptive and cooperative car-following”. PhD thesis. Université de recherche Paris Sciences et Lettres PSL Research University, 2021.

[105] Martin Treiber and Dirk Helbing. “Realistische Mikrosimulation von Strassenverkehr mit einem einfachen Modell”. In: *16th Symposium Simulationstechnik ASIM*. Vol. 2002. 2002, p. 80.

[106] Wouter J Schakel, Bart Van Arem, and Bart D Netten. “Effects of cooperative adaptive cruise control on traffic flow stability”. In: *13th International IEEE Conference on Intelligent Transportation Systems*. IEEE. 2010, pp. 759–764.

[107] DVAHG Swaroop, J Karl Hedrick, CC Chien, and Petros Ioannou. “A comparision of spacing and headway control laws for automatically controlled vehicles1”. In: *Vehicle system dynamics* 23.1 (1994), pp. 597–625.

[108] Hamed Rezaee, Thomas Parisini, and Marios M Polycarpou. “Leaderless Cooperative Adaptive Cruise Control Based on Constant Time-Gap Spacing Policy”. In: *IEEE Transactions on Automatic Control* (2023).

[109] Lei Zuo, Peng Wang, Maode Yan, and Xu Zhu. “Platoon tracking control with road-friction based spacing policy for nonlinear vehicles”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.11 (2022), pp. 20810–20819.

[110] Paul Wijnbergen, Mark Jeeninga, and Bart Besselink. “Nonlinear spacing policies for vehicle platoons: A geometric approach to decentralized control”. In: *Systems & Control Letters* 153 (2021), p. 104954.

[111] Aurelio González-Villaseñor, Alasdair C Renfrew, and Paul J Brunn. "A controller design methodology for close headway spacing strategies for automated vehicles". In: *International Journal of Control* 80.2 (2007), pp. 179–189.

[112] Chi-Ying Liang and Huei Peng. "String stability analysis of adaptive cruise controlled vehicles". In: *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing* 43.3 (2000), pp. 671–677.

[113] Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. "Lp string stability of cascaded systems: Application to vehicle platooning". In: *IEEE Transactions on Control Systems Technology* 22.2 (2013), pp. 786–793.

[114] Elaine Shaw and J Karl Hedrick. "Controller design for string stable heterogeneous vehicle strings". In: *2007 46th IEEE Conference on Decision and Control*. IEEE. 2007, pp. 2868–2875.

[115] Ali Balador, Alessandro Bazzi, Unai Hernandez-Jayo, Idoia de la Iglesia, and Hossein Ahmadvand. "A survey on vehicular communication for cooperative truck platooning application". In: *Vehicular Communications* 35 (2022), p. 100460.

[116] Carlos Flores, Vicente Milanés, and Fawzi Nashashibi. "Online feedforward/feedback structure adaptation for heterogeneous CACC strings". In: *2018 Annual American Control Conference (ACC)*. IEEE. 2018, pp. 49–55.

[117] Joshué Pérez, Vicente Milanés, Jorge Godoy, Jorge Villagra, and Enrique Onieva. "Cooperative controllers for highways based on human experience". In: *Expert Systems with Applications* 40.4 (2013), pp. 1024–1033.

[118] Alex Gunagwera. "Autonomous vehicle platoon modeling and control using PID and linear quadratic regulator". In: (2022).

[119] Jicheng Chen, Henglai Wei, Hui Zhang, and Yang Shi. "Asynchronous self-triggered stochastic distributed mpc for cooperative vehicle platooning over vehicular ad-hoc networks". In: *IEEE Transactions on Vehicular Technology* (2023).

[120] Hong Wang, Li-Ming Peng, Zichun Wei, et al. "A holistic robust motion control framework for autonomous platooning". In: *IEEE Transactions on Vehicular Technology* (2023).

[121] Ajay G Iyer, Jagannath Samantaray, Samsaptak Ghosh, Arnab Dey, and Sohom Chakrabarty. "Sliding Mode Control Using Power Rate Exponential Reaching Law for Urban Platooning". In: *IFAC-PapersOnLine* 55.1 (2022), pp. 516–521.

[122] Francisco Navas, Vicente Milanés, Carlos Flores, and Fawzi Nashashibi. "Multi-model adaptive control for CACC applications". In: *IEEE Transactions on Intelligent Transportation Systems* 22.2 (2020), pp. 1206–1216.

[123] Justin M Kennedy, Julian Heinovski, Daniel E Quevedo, and Falko Dressler. "Centralized Model Predictive Control with Human-Driver Interaction for Platooning". In: *IEEE Transactions on Vehicular Technology* (2023).

[124] Carlos Hidalgo, Ray Lattarulo, Carlos Flores, and Joshué Pérez Rastelli. "Platoon merging approach based on hybrid trajectory planning and CACC strategies". In: *Sensors* 21.8 (2021), p. 2626.

[125] Shaohua Cui, Yongjie Xue, Kun Gao, Maolong Lv, and Bin Yu. "Adaptive Collision-Free Trajectory Tracking Control for String Stable Bidirectional Platoons". In: *IEEE Transactions on Intelligent Transportation Systems* (2023).

[126] Shixi Wen and Ge Guo. "Vehicular Platoon Control with Gain Variations via Unreliable VANETs". In: *IEEE Transactions on Vehicular Technology* (2023).

[127] Elham Abolfazli, Wei Jiang, and Themistoklis Charalambous. “Towards Establishing String Stability Conditions For Heterogeneous Vehicle Platoons Under the MPF Topology”. In: *2022 European Control Conference (ECC)*. IEEE. 2022, pp. 944–950.

[128] Wei Wu, Yang Liu, Wei Liu, et al. “Autonomous intersection management for connected and automated vehicles: A lane-based method”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.9 (2021), pp. 15091–15106.

[129] Lianzhen Wei, Zirui Li, Jianwei Gong, Cheng Gong, and Jiachen Li. “Autonomous driving strategies at intersections: Scenarios, state-of-the-art, and future outlooks”. In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 44–51.

[130] Can Zhao, Li Li, Xin Pei, et al. “A comparative study of state-of-the-art driving strategies for autonomous vehicles”. In: *Accident Analysis & Prevention* 150 (2021), p. 105937.

[131] Marwan Salim Mahmood Al-Dabbagh, Ali Al-Sherbaz, and Scott Turner. “The impact of road intersection topology on traffic congestion in urban cities”. In: *Intelligent Systems and Applications: Proceedings of the 2018 Intelligent Systems Conference (IntelliSys) Volume 1*. Springer. 2019, pp. 1196–1207.

[132] Fushi Lian, Bokui Chen, Kai Zhang, et al. “Adaptive traffic signal control algorithms based on probe vehicle data”. In: *Journal of Intelligent Transportation Systems* 25.1 (2021), pp. 41–57.

[133] Soheil Mohamad Alizadeh Shabestary and Baher Abdulhai. “Deep learning vs. discrete reinforcement learning for adaptive traffic signal control”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2018, pp. 286–293.

[134] SYED MASIUR Rahman and NEDAL T Ratrout. “Review of the fuzzy logic based approach in traffic signal control: Prospects in Saudi Arabia”. In: *Journal of transportation Systems engineering and information Technology* 9.5 (2009), pp. 58–70.

[135] Md Abdus Samad Kamal, Jun-ichi Imura, Tomohisa Hayakawa, Akira Ohata, and Kazuyuki Aihara. “A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights”. In: *IEEE Transactions on Intelligent Transportation Systems* 16.3 (2014), pp. 1136–1147.

[136] Weilong Song, Guangming Xiong, and Huiyan Chen. “Intention-aware autonomous driving decision-making in an uncontrolled intersection.” In: *Mathematical Problems in Engineering* (2016).

[137] Lijun Qian, Chen Chen, Bing Wu, Liang Xuan, and Tianxiang Wang. “Optimal control of connected and automated vehicles at unsignalized intersections: Discrete and regroup”. In: *2020 International Conference on Intelligent Engineering and Management (ICIEM)*. IEEE. 2020, pp. 370–375.

[138] Bai Li, Youmin Zhang, Yue Zhang, Ning Jia, and Yuming Ge. “Near-optimal online motion planning of connected and automated vehicles at a signal-free and lane-free intersection”. In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2018, pp. 1432–1437.

[139] Xuemei Chen, Yufan Sun, Yangjiaxin Ou, et al. “A conflict decision model based on game theory for intelligent vehicles at urban unsignalized intersections”. In: *IEEE Access* 8 (2020), pp. 189546–189555.

[140] Yue Zhang, Andreas A Malikopoulos, and Christos G Cassandras. “Decentralized optimal control for connected automated vehicles at intersections including left and right turns”. In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE. 2017, pp. 4428–4433.

[141] Marcel Walch, Marcel Woide, Kristin Mühl, Martin Baumann, and Michael Weber. "Cooperative overtaking: Overcoming automated vehicles' obstructed sensor range via driver help". In: *Proceedings of the 11th international conference on automotive user interfaces and interactive vehicular applications*. 2019, pp. 144–155.

[142] Anna-Maria Sourelli. "Exploring Acceptable Overtaking Behaviour for Automated Vehicles". PhD thesis. Loughborough University, 2021.

[143] Shikhar Singh Lodh, Neetesh Kumar, and Pradumn Kumar Pandey. "Autonomous Vehicular Overtaking Maneuver: A Survey and Taxonomy". In: *Vehicular Communications* (2023), p. 100623.

[144] Ray Lattarulo and Joshué Pérez Rastelli. "A hybrid planning approach based on MPC and parametric curves for overtaking maneuvers". In: *Sensors* 21.2 (2021), p. 595.

[145] Xiaoyu Yang and Huiyun Li. "Model Predictive Motion Planning for Autonomous Vehicle in Mid-high Overtaking Scene". In: *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE. 2020, pp. 1–5.

[146] Aneesh Raghavan, Jieqiang Wei, John S Baras, and Karl Henrik Johansson. "Stochastic control formulation of the car overtake problem". In: *IFAC-PapersOnLine* 51.9 (2018), pp. 124–129.

[147] Daofei Li and Hao Pan. "Two-lane two-way overtaking decision model with driving style awareness based on a game-theoretic framework". In: *Transportmetrica A: transport science* 19.3 (2023), p. 2076755.

[148] Meihong Zhang, Tingting Zhang, Liuqing Yang, Hongguang Xu, and Qinyu Zhang. "An autonomous overtaking maneuver based on relative position information". In: *Journal of Communications and Information Networks* 4.2 (2019), pp. 101–110.

[149] Seungmin Jeon, Kibeom Lee, and Dongsuk Kum. "Overtaking decision and trajectory planning in highway via hierarchical architecture of conditional state machine and chance constrained model predictive control". In: *Robotics and Autonomous Systems* 151 (2022), p. 104014.

[150] Cosmin Ginerica, Mihai Zaha, Florin Gogianu, et al. "Observenet control: A vision-dynamics learning approach to predictive control in autonomous vehicles". In: *IEEE Robotics and Automation Letters* 6.4 (2021), pp. 6915–6922.

[151] Xiaoxiang Li, Xinyou Qiu, Jian Wang, and Yuan Shen. "A deep reinforcement learning based approach for autonomous overtaking". In: *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2020, pp. 1–5.

[152] Sahaya Beni Prathiba, Gunasekaran Raja, and Neeraj Kumar. "Intelligent cooperative collision avoidance at overtaking and lane changing maneuver in 6G-V2X communications". In: *IEEE Transactions on Vehicular Technology* 71.1 (2021), pp. 112–122.

[153] Anna-Maria Sourelli, Ruth Welsh, and Pete Thomas. "Objective and perceived risk in overtaking: The impact of driving context". In: *Transportation research part F: traffic psychology and behaviour* 81 (2021), pp. 190–200.

[154] Rafael Toledo-Moreo, José Santa, and M Zamora-Izquierdo. "A cooperative overtaking assistance system". In: *Planning, Perception and Navigation for Intelligent Vehicles (PPNIV)* 50 (2009).

[155] Jan Cedric Mertens, Jürgen Hauenstein, Frank Diermeyer, Lennard Jahn, and Sven Kraus. "Cooperative Truck Overtaking on Freeways". In: *2020 Fifteenth International Conference on Ecological Vehicles and Renewable Energies (EVER)*. IEEE. 2020, pp. 1–15.

[156] Martin Strunz, Julian Heinovski, and Falko Dressler. "CoOP: V2V-based cooperative overtaking for platoons on freeways". In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 1090–1097.

[157] Omar Nassef, Luis Sequeira, Elias Salam, and Toktam Mahmoodi. “Building a lane merge coordination for connected vehicles using deep reinforcement learning”. In: *IEEE Internet of Things Journal* 8.4 (2020), pp. 2540–2557.

[158] Alessandro Falsone, Beatrice Melani, and Maria Prandini. “Lane change in automated driving: an explicit coordination strategy”. In: *IEEE Control Systems Letters* 7 (2022), pp. 205–210.

[159] Tianyu Shi, Pin Wang, Xuxin Cheng, Ching-Yao Chan, and Ding Huang. “Driving decision and control for automated lane change behavior based on deep reinforcement learning”. In: *2019 IEEE intelligent transportation systems conference (ITSC)*. IEEE. 2019, pp. 2895–2900.

[160] ASM Bakibillah, Md Abdus Samad Kamal, Chee Pin Tan, et al. “Bi-Level coordinated merging of connected and automated vehicles at roundabouts”. In: *Sensors* 21.19 (2021), p. 6533.

[161] Mohamad Hafizulazwan Bin Mohamad Nor and Toru Namerikawa. “Merging of connected and automated vehicles at roundabout using model predictive control”. In: *2018 57th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*. IEEE. 2018, pp. 272–277.

[162] Vicente Milanés, Jorge Godoy, Jorge Villagrá, and Joshué Pérez. “Automated on-ramp merging system for congested traffic situations”. In: *IEEE Transactions on Intelligent Transportation Systems* 12.2 (2010), pp. 500–508.

[163] Guofa Li, Weiyan Zhou, Siyan Lin, Shen Li, and Xingda Qu. “On-Ramp Merging for Highway Autonomous Driving: An Application of a New Safety Indicator in Deep Reinforcement Learning”. In: *Automotive Innovation* (2023), pp. 1–13.

[164] Joshué Pérez Rastelli and Matilde Santos Peñas. “Fuzzy logic steering control of autonomous vehicles inside roundabouts”. In: *Applied Soft Computing* 35 (2015), pp. 662–669.

[165] Joshué Pérez, Vicente Milanés, Teresa De Pedro, and Ljubo Vlacic. “Autonomous driving manoeuvres in urban road traffic environment: a study on roundabouts”. In: *IFAC Proceedings Volumes* 44.1 (2011), pp. 13795–13800.

[166] Hang Cao and Mate Zoldy. “MPC tracking controller parameters impacts in roundabouts”. In: *Mathematics* 9.12 (2021), p. 1394.

[167] David González, Joshué Pérez, and Vicente Milanés. “Parametric-based path generation for automated vehicles at roundabouts”. In: *Expert Systems with Applications* 71 (2017), pp. 332–341.

[168] Júnior AR Silva and Valdir Grassi. “Path planning at roundabouts using piecewise linear continuous curvature curves”. In: *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*. IEEE. 2017, pp. 1–6.

[169] Larissa Labakhua, Urbano Nunes, Rui Rodrigues, and Fátima S Leite. “Smooth trajectory planning for fully automated passengers vehicles: Spline and clothoid based methods and its simulation”. In: *Informatics in Control Automation and Robotics: Selected Papers from the International Conference on Informatics in Control Automation and Robotics 2006*. Springer. 2008, pp. 169–182.

[170] Carlos Hidalgo, Ray Lattarulo, Joshué Pérez, and Estibaliz Asua. “Intelligent Longitudinal Merging Maneuver at Roundabouts Based on Hybrid Planning Approach”. In: *Computer Aided Systems Theory–EUROCAST 2019: 17th International Conference, Las Palmas de Gran Canaria, Spain, February 17–22, 2019, Revised Selected Papers, Part II 17*. Springer. 2020, pp. 129–136.

[171] Ran Tian, Sisi Li, Nan Li, et al. “Adaptive game-theoretic decision making for autonomous vehicle control at roundabouts”. In: *2018 IEEE Conference on Decision and Control (CDC)*. IEEE. 2018, pp. 321–326.

[172] Stefano Masi, Philippe Xu, and Philippe Bonnifait. “Adapting the virtual platooning concept to roundabout crossing”. In: *2018 IEEE intelligent vehicles symposium (IV)*. IEEE. 2018, pp. 1366–1372.

[173] Junjie Wang, Qichao Zhang, Dongbin Zhao, and Yaran Chen. “Lane change decision-making through deep reinforcement learning with rule-based constraints”. In: *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2019, pp. 1–6.

[174] Eleonora Andreotti, Maytheewat Aramrattana, et al. “Cooperative Merging Strategy Between Connected Autonomous Vehicles in Mixed Traffic”. In: *IEEE Open Journal of Intelligent Transportation Systems* 3 (2022), pp. 825–837.

[175] Haihua Zhu, Yi Zhang, Changchun Liu, and Wei Shi. “An Adaptive Reinforcement Learning-Based Scheduling Approach with Combination Rules for Mixed-Line Job Shop Production”. In: *Mathematical Problems in Engineering* 2022 (2022).

[176] Jackeline Rios-Torres and Andreas A Malikopoulos. “Automated and cooperative vehicle merging at highway on-ramps”. In: *IEEE Transactions on Intelligent Transportation Systems* 18.4 (2016), pp. 780–789.

[177] Bin Ran, Shawn Leight, and Ben Chang. “A microscopic simulation model for merging control on a dedicated-lane automated highway system”. In: *Transportation Research Part C: Emerging Technologies* 7.6 (1999), pp. 369–388.

[178] Hangxu Ji, Gang Wu, Yuhai Zhao, Ye Yuan, and Guoren Wang. “Multi-job Merging Framework and Scheduling Optimization for Apache Flink”. In: *Database Systems for Advanced Applications: 26th International Conference, DASFAA 2021, Taipei, Taiwan, April 11–14, 2021, Proceedings, Part I 26*. Springer. 2021, pp. 20–36.

[179] Jie Ni, Jingwen Han, and Fei Dong. “Multivehicle cooperative lane change control strategy for intelligent connected vehicle”. In: *Journal of Advanced Transportation* 2020 (2020), pp. 1–10.

[180] Zhouqiao Zhao, Ziran Wang, Guoyuan Wu, Fei Ye, and Matthew J Barth. “The state-of-the-art of coordinated ramp control with mixed traffic conditions”. In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2019, pp. 1741–1748.

[181] Hao Liu, Xingan Kan, Steven E Shladover, Xiao-Yun Lu, and Robert E Ferlis. “Impact of cooperative adaptive cruise control on multilane freeway merge capacity”. In: *Journal of Intelligent Transportation Systems* 22.3 (2018), pp. 263–275.

[182] Chaojie Wang, Yu Wang, and Srinivas Peeta. “Cooperative roundabout control strategy for connected and autonomous vehicles”. In: *Applied Sciences* 12.24 (2022), p. 12678.

[183] Ali Irshayyid and Jun Chen. “Comparative Study of Cooperative Platoon Merging Control Based on Reinforcement Learning”. In: *Sensors* 23.2 (2023), p. 990.

[184] Anirudh Paranjothi, Mohammed Atiquzzaman, and Mohammad S Khan. “PMCD: Platoon-Merging approach for cooperative driving”. In: *Internet Technology Letters* 3.1 (2020), e139.

[185] Huan Gao, Yanqing Cen, Bo Liu, et al. “A collaborative merging method for connected and automated vehicle platoons in a freeway merging area with considerations for safety and efficiency”. In: *Sensors* 23.9 (2023), p. 4401.

[186] Qianwen Li, Zhiwei Chen, and Xiaopeng Li. “A review of connected and automated vehicle Platoon merging and splitting operations”. In: *IEEE Transactions on Intelligent Transportation Systems* (2022).

[187] Ray Lattarulo, Joshué Pérez, and Martin Dendaluce. “A complete framework for developing and testing automated driving controllers”. In: *IFAC-PapersOnLine* 50.1 (2017), pp. 258–263.

[188] Carlos Hidalgo, Myriam Vaca, Mateusz P Nowak, et al. “Detection, control and mitigation system for secure vehicular communication”. In: *Vehicular Communications* 34 (2022), p. 100425.

[189] Joanna Domanska, Erol Gelenbe, Tadek Czachorski, Anastasis Drosou, and Dimitrios Tzovaras. “Research and innovation action for the security of the internet of things: The SerIoT project”. In: *Security in Computer and Information Sciences: First International ISCIS Security Workshop 2018, Euro-CYBERSEC 2018, London, UK, February 26-27, 2018, Revised Selected Papers 1*. Springer International Publishing. 2018, pp. 101–118.

[190] Erol Gelenbe and Mert Nakip. “Real-Time Cyberattack Detection with Offline and Online Learning”. In: *2023 IEEE 29th International Symposium on Local and Metropolitan Area Networks (LANMAN)*. IEEE. 2023, pp. 1–6.

[191] Miltiadis Siavvas, Ilias Kalouptsoglou, Dimitrios Tsoukalas, and Dionysios Kehagias. “A self-adaptive approach for assessing the criticality of security-related static analysis alerts”. In: *Computational Science and Its Applications–ICCSA 2021: 21st International Conference, Cagliari, Italy, September 13–16, 2021, Proceedings, Part VII 21*. Springer. 2021, pp. 289–305.

[192] The MathWorks Inc. *MATLAB version: 9.13.0 (R2020b)*. Natick, Massachusetts, United States, 2020.

[193] Boris Houska, Hans Joachim Ferreau, and Moritz Diehl. “ACADO toolkit—An open-source framework for automatic control and dynamic optimization”. In: *Optimal Control Applications and Methods* 32.3 (2011), pp. 298–312.

[194] Javier Cuadrado, David Vilela, Iñaki Iglesias, Adrián Martín, and Alberto Peña. “A multibody model to assess the effect of automotive motor in-wheel configuration on vehicle stability and comfort”. In: *ECCOMAS Multibody Dynamics* (2013).

[195] A. Parra, A. J. Rodríguez, A. Zubizarreta, and J. Pérez. “Validation of a Real-Time Capable Multibody Vehicle Dynamics Formulation for Automotive Testing Frameworks Based on Simulation”. In: *IEEE Access* 8 (2020), pp. 213253–213265.

[196] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. “CARLA: An Open Urban Driving Simulator”. In: *arXiv preprint arXiv:1711.03938* (2017).

[197] Shinpei Kato. *“Autoware*. 2017.

[198] Ray Lattarulo, Carlos Hidalgo, Asier Arizala, and Joshue Perez. “Audric2: A modular and highly interconnected automated driving framework focus on decision making and vehicle control”. In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 763–769.

[199] C Hidalgo, M Marcano, G Fernández, and JM Pérez. “Maniobras cooperativas aplicadas a vehículos automatizados en entornos virtuales y reales”. In: *Revista Iberoamericana de Automática e Informática industrial* 17.1 (2020), pp. 56–65.

[200] Darbha Swaroop and KR Rajagopal. “A review of constant time headway policy for automatic vehicle following”. In: *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 01TH8585)*. IEEE. 2001, pp. 65–69.

[201] Vicente Milanés and Steven E Shladover. “Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data”. In: *Transportation Research Part C: Emerging Technologies* 48 (2014), pp. 285–300.

[202] Carlos Flores, Vicente Milanés, and Fawzi Nashashibi. “Using fractional calculus for cooperative car-following control”. In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2016, pp. 907–912.

[203] Carlos Flores, Pierre Merdrignac, Raoul de Charette, et al. “A cooperative car-following/emergency braking system with prediction-based pedestrian avoidance capabilities”. In: *IEEE Transactions on Intelligent Transportation Systems* 20.5 (2018), pp. 1837–1846.

[204] Ray Lattarulo, Leonardo González, Enrique Martí, et al. “Urban motion planning framework based on n-bézier curves considering comfort and safety”. In: *Journal of Advanced Transportation* 2018 (2018).

[205] Jorge Villagra, Vicente Milanés, Joshué Pérez, and Jorge Godoy. “Smooth path and speed planning for an automated public transport vehicle”. In: *Robotics and Autonomous Systems* 60.2 (2012), pp. 252–265.

[206] Fabian Poggenhans, Jan-Hendrik Pauls, Johannes Janosovits, et al. “Lanelet2: A high-definition map framework for the future of automated driving”. In: *2018 21st international conference on intelligent transportation systems (ITSC)*. IEEE. 2018, pp. 1672–1679.

[207] Jiyo Palatti, Andrei Aksjonov, Gokhan Alcan, and Ville Kyrki. “Planning for safe abortable overtaking maneuvers in autonomous driving”. In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2021, pp. 508–514.

[208] Mengxuan Zhang, Nan Li, Anouck Girard, and Ilya Kolmanovsky. “A finite state machine based automated driving controller and its stochastic optimization”. In: *Dynamic Systems and Control Conference*. Vol. 58288. American Society of Mechanical Engineers. 2017, V002T07A002.

[209] Ezequiel Gonzale Debada and Denis Gillet. “Virtual vehicle-based cooperative maneuver planning for connected automated vehicles at single-lane roundabouts”. In: *IEEE Intelligent Transportation Systems Magazine* 10.4 (2018), pp. 35–46.

[210] Hansung Kim and Francesco Borrelli. “Facilitating Cooperative and Distributed Multi-Vehicle Lane Change Maneuvers”. In: *arXiv preprint arXiv:2301.04316* (2023).

[211] Xu Han, Runsheng Xu, Xin Xia, et al. “Strategic and tactical decision-making for cooperative vehicle platooning with organized behavior on multi-lane highways”. In: *Transportation Research Part C: Emerging Technologies* 145 (2022), p. 103952.

[212] Ray Lattarulo and Joshue Perez. “Fast real-time trajectory planning method with 3rd-order curve optimization for automated vehicles”. In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2020, pp. 1–6.

[213] Ray Lattarulo, Leonardo González, and Joshue Perez. “Real-time trajectory planning method based on n-order curve optimization”. In: *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE. 2020, pp. 751–756.

[214] Michael JD Powell et al. “The BOBYQA algorithm for bound constrained optimization without derivatives”. In: *Cambridge NA Report NA2009/06, University of Cambridge, Cambridge* 26 (2009).

[215] B Atanassow and K Sjöberg. “Platooning protocol definition and Communication strategy”. In: *H2020-Project ENSEMBLE Deliverable D* 2 (2018), p. 8.

[216] *Intelligent Transport Systems – Vehicular Communications – GeoNetworking – Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications*. Standard. European Telecommunications Standards Institute., Mar. 2000.

[217] Carlos Hidalgo, Ray Lattarulo, Joshué Pérez, and Estibaliz Asua. “Hybrid trajectory planning approach for roundabout merging scenarios”. In: *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE. 2019, pp. 1–6.

[218] Mátyás Szántó, Carlos Hidalgo, Leonardo González, et al. “Trajectory Planning of Automated Vehicles Using Real-Time Map Updates”. In: *IEEE Access* (2023).

[219] Halil Beglerovic, Abhishek Ravi, Niklas Wikström, et al. “Model-based safety validation of the automated driving functio highway pilot”. In: *8th International Munich Chassis Symposium 2017: chassis. tech plus*. Springer. 2017, pp. 309–329.

[220] Tomer Toledo and David Zohar. “Modeling duration of lane changes”. In: *Transportation Research Record* 1999.1 (2007), pp. 71–78.