



HAL
open science

Collective intelligence strategies for efficient autonomous industrial vehicles

Juliette Grosset

► **To cite this version:**

Juliette Grosset. Collective intelligence strategies for efficient autonomous industrial vehicles. Operations Research [math.OA]. Ecole nationale supérieure Mines-Télécom Atlantique, 2024. English. NNT : 2024IMTA0436 . tel-04921399

HAL Id: tel-04921399

<https://theses.hal.science/tel-04921399v1>

Submitted on 30 Jan 2025

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

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

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE
MINES-TÉLÉCOM ATLANTIQUE BRETAGNE
PAYS DE LA LOIRE – IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 648
Sciences pour l'Ingénieur et le Numérique
Spécialité : *Informatique*

Par

Juliette GROSSET

Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles

Thèse présentée et soutenue à IMT Atlantique, Rennes, le 26 novembre 2024

Unité de recherche : IRISA

Thèse N° : 2024IMTA0436

Rapporteurs avant soutenance :

Leila MERGHEM-BOULAHIA Professeure, Université de Technologie de Troyes
Gérard CHALHOUB Professeur, Université Clermont Auvergne

Composition du Jury :

Présidente :	Maroua BOUZID	Professeure, Université de Caen Normandie
Examineurs :	Leila MERGHEM-BOULAHIA	Professeure, Université de Technologie de Troyes
	Gérard CHALHOUB	Professeur, Université Clermont Auvergne
	Cédric BUCHE	Professeur, ENIB, Naval Group Pacific (Adelaide, Australie)
Dir. de thèse :	Jean-Marie BONNIN	Professeur, IMT Atlantique
Co-dir. de thèse :	Alain-Jérôme FOUGÈRES	Professeur, ECAM Louis de Broglie
Encadr. de thèse :	Moïse DJOKO-KOUAM	Enseignant-Chercheur, ECAM Louis de Broglie

Invité(s) :

Christophe COUTURIER Ingénieur, iRider

REMERCIEMENTS

La thèse marque l'aboutissement d'une aventure de trois années, riches en expériences, en rencontres et en apprentissages. Une aventure à la fois intense et exigeante, que je n'aurais pas pu mener seule ! C'est donc ici, dans ces quelques pages de remerciements, avec une émotion profonde et beaucoup de sincérité, que je souhaite exprimer ma gratitude et l'impact inestimable de toutes les personnes qui m'ont accompagnée, soutenue et inspirée tout au long de ce parcours.

Tout d'abord, je tiens à remercier l'ensemble des membres du jury pour avoir accepté d'évaluer mon manuscrit, ma soutenance et pour l'intérêt qu'ils ont porté à mes travaux de recherche. Merci à Leïla Merghem-Boulahia, Maroua Bouzid, Gérard Chalhoub et Cédric Buche pour leur temps et leurs précieux commentaires.

Merci Vincent Drevelle et Frédéric Weis pour avoir accepté d'être membres de mon comité de thèse, pour avoir suivi mes travaux et pour leurs conseils et orientations lors des réunions annuelles.

Je souhaite exprimer ma gratitude à la région Bretagne et à l'ECAM Louis de Broglie pour le co-financement de ce projet de thèse. Un grand merci également à l'IMT Atlantique, qui m'a accueillie au sein de l'équipe EASE de l'IRISA.

Merci infiniment à mes directeurs de thèse, Alain-Jérôme Fougères et Jean-Marie Bonnin. Votre confiance et votre supervision ont été des piliers essentiels tout au long de cette aventure. Nos réunions mensuelles, toujours enrichissantes et agréables, ont été des moments précieux. Merci pour votre disponibilité, votre écoute attentive, vos conseils avisés et tout le soutien que vous m'avez apporté durant ces trois années. Je ne pourrai jamais assez vous remercier pour votre patience et votre bienveillance après ma blessure à la cheville. Votre compréhension a été pour moi très importante. Travailler avec vous a été une expérience profondément épanouissante, et j'espère sincèrement que nous aurons l'occasion de collaborer à nouveau à l'avenir !

Ces remerciements s'étendent également à mon encadrant de thèse, Moïse Djoko-Kouam, avec qui j'ai pu notamment partager de nombreux débats sur la robotique, l'encadrement de projets et de stagiaires à l'ECAM. Merci pour l'autonomie et la confiance que tu m'as accordée.

Merci Christophe Couturier, seulement 1 an d'encadrement, avant que tu ne partes pour de nouveaux horizons professionnels, mais je n'en oublie pas moins tes idées, et ta disponibilité jusqu'au bout de ma thèse en acceptant d'être membre invité de mon jury pour mon plus grand plaisir.

Un immense merci à Jean-Marie et Manabu Tsukada (Tsukada sensei) pour m'avoir offert l'opportunité de partir en mobilité au laboratoire Tsukada Laboratory. Ces trois mois passés au Japon, à l'université de Tokyo, ont été une expérience inoubliable, riche en découvertes et en rencontres. Merci Manabu, pour ton accueil chaleureux, un cadre de travail enrichissant et pour m'avoir permis de travailler avec toi.

Le temps passé entre mes deux bureaux, à l'ECAM et à l'IMT Atlantique, m'ont offert un environnement de travail idéal, à la fois convivial, stimulant et enrichissant. Cette aventure aurait été bien différente sans toutes les personnes qui m'ont entourée au quotidien.

À l'IMT Atlantique, un merci tout particulier à l'équipe EASE, qui était comme une famille. Merci à mes co-bureaux Elodie, Mariam, Abdoul-Majid, et à Laudin qui était là à mes débuts. Une mention spéciale à Elodie, avec qui j'ai notamment partagé mon tout premier jour à l'IMT Atlantique, et le premier séminaire SRCO où nous ne connaissions personne. Et bien sûr, je n'oublie pas le fameux bureau voisin : Hassan, Hadi, et Maxime. Maxime, Washington avec toi, restera gravé, on aura quand même tapé du pied sur la rivière Potomac, si c'est pas dingue ça ! Une pensée très forte à Hassan, Mariam et Maxime, qui sont arrivés après moi : je vous souhaite tout le succès que vous méritez dans cette belle aventure. Docteur Hadi, je te souhaite tout le meilleur pour tes nouveaux projets.

Un immense merci aux permanents Jean-Marie, Christophe, Frédéric et Paul, qui m'ont accueillie avec bienveillance et m'ont intégrée à l'équipe dès mon arrivée. Votre disponibilité et votre soutien ont été précieux tout au long de ce parcours. Je garderai toujours en mémoire les pauses café qui s'étiraient, les déjeuners partagés et ces moments de convivialité. Un souvenir particulier restera gravé : le séminaire sur l'Île d'Arz, un moment hors du temps rempli d'échanges et de complicité, merci Gwenaëlle pour l'organisation.

Mes remerciements s'étendent aussi à l'ensemble de mes collègues de l'ADER, au-delà de l'équipe EASE. Merci à Léo, Gwen, Pierre-Marie, Loïc, Antoine, Modou, Awaleh et Amina pour avoir rendu ce voyage plus riche et plus humain. Merci pour ces discussions animées au centre de vie, ces moments au bar, les soirées à l'escalade avec Antoine, Léo et

Pierre-Marie, les sorties en Mach 6.50 avec Pierre-Marie, les parties de foot avec Awaleh, Renzo et Modou, ou encore les soirées au stade avec Renzo et Loïc. Et que dire des débats centrées sur le sport et les péripéties du Stade Rennais avec Renzo, Loutfi et Jean-Marie. Loïc nous rejoignait parfois, mais seulement si Strasbourg était en bonne position au classement ! Après le déjeuner, les parties d'échecs avec Hadi devenaient presque un rituel, souvent ponctuées par le regard désespéré – mais amusé – de Georgios face à notre niveau. Que de rires et de bons moments !

Un merci spécial à Antoine et Pierre-Marie, mes compagnons des débuts, avec qui j'ai partagé tant de souvenirs. Je vous souhaite le meilleur pour vos projets à venir. Et à Léo, Gwen et Loïc, merci infiniment pour votre aide précieuse et vos nombreux conseils tout au long de cette aventure.

Enfin, un grand merci à l'ensemble du département SRCD : Géraldine, Hélène, Xavier, Baptiste, Briac, Laurent, Patrick, Nicolas, Romaric, Georgios, Loutfi, Alexander... Et ceux qui s'occupent du département : Delphine, Catherine, Franck, Vincent. Avec une mention spéciale pour Sandrine : merci pour ta disponibilité, ta réactivité et ton aide inestimable pour toutes mes démarches administratives.

A l'ECAM, je n'étais pas moins bien entourée ! Tout d'abord, un merci exceptionnel à Victorien, avec qui j'ai partagé le fameux bureau S205. Merci pour ta bonne humeur, tous ces moments de partage, les pauses-café, les soirées, ces wordle/worldle/tusmo à n'en plus finir ! Le meilleur pour la suite de ta thèse, et pour tes projets. A ce premier été à l'ECAM, en compagnie des deux stagiaires préférés, Léa et Antoine devenu grand aujourd'hui ! Une pensée pour Paul, avec qui j'ai commencé cette aventure de thèse, et à Raphaël, le petit dernier doctorant, devenu un très bon ami aujourd'hui ! Je vous souhaite le meilleur.

Je tiens à remercier chaleureusement tous les collègues du département informatique pour leur accueil et leur intégration : Alain-Jérôme, Moïse, Denys, Jean-Marc, Hervé, Maryem, Noussaïba, Phuong, Ouzna et Van-Hieu. Merci également à tous les autres collègues de l'ECAM : Céline, Hélène, Nicolas, Mayssa, Jean-François, les Aurélie, Camille, Céline, Elise, Gwendaëlle, Benjamin, Ahmad, Khaled, Etienne, Johan, Tabi, Mélanie, Jelle, Florence. Merci à François, Guillaume et Clément, qui m'ont aidé à mettre en place mes expérimentations. Merci également à Aude et Thomas pour la communication pour mettre en avant nos recherches.

Un merci tout particulier à Katia, une assistante des départements et de la recherche au top. Merci pour ta disponibilité, ta bienveillance, ton écoute et ton soutien moral

permanent ! Une mention spéciale pour Chrystèle Martin, merci pour ton soutien et de m'avoir suivie jusqu'à la fin de ma thèse, même après ton départ.

Ensuite, je souhaite remercier tous mes amis. Vous m'avez permis de vivre des moments incroyables, vous m'avez écouté et pour cela, je vous en suis éternellement reconnaissante : Alexis, Zoé, Léa, Antoine, Valentin M., Victorien, Raphaël, Luc, François, Auguste, Bastien, Damien, Maëlle, Gireg, Anwar, Agathe, Louise, Hugo, Julie, Victor, Camille, Benjamin, Cécile, Alex, Thibaud, Lise, Evan, Claudia et Margaux. Une aventure à MT180, avec Agathe et Louise que je n'oublierai pas.

Mention spéciale à Luc et Zoé pour votre aide précieuse et votre disponibilité sans faille. Cette aventure a été marquée par un moment difficile de ma vie, ma blessure à la cheville. Merci du fond du cœur à toutes les personnes qui m'ont soutenu durant cette période, en particulier Coco, Luc, Nolwen et Valentin R.

Un remerciement tout particulier à ma famille, vous êtes mes repères et ma force au quotidien. Merci Maman, Mamie, Mickaël, Iris, Manuel, Miguel, Sylvie, Alain, Antoine et Manon. Merci pour vos encouragements, pour toutes les valeurs que vous m'avez transmises et qui font de moi la personne que je suis aujourd'hui. Merci d'avoir toujours eu confiance en moi. Bien évidemment, je ne t'oublie pas Coco, tu as été un pilier essentiel tout au long de cette aventure, et je ne pourrai jamais assez te remercier pour tout ce que tu fais pour moi. Merci d'être toi, merci pour tout. Merci Sylvie, Rémi et Laurène, ma deuxième famille, pour tout votre soutien, votre présence le jour de la soutenance m'a fait le plus grand plaisir.

Enfin, je tiens à remercier chaleureusement toutes les personnes qui étaient présentes lors de ma soutenance de thèse : ma famille, mes collègues et mes amis. Certains d'entre vous ont fait le déplacement depuis loin, ont passé quelques jours à l'appartement pour me soutenir ce jour-là, et je ne l'oublierai jamais, vraiment merci. C'était un moment très fort pour moi, voir le petit amphi de l'IMT Atlantique rempli, avec même des personnes debout, m'a profondément émue. Cette image restera gravée dans ma mémoire. Un grand merci également à ceux qui n'ont pas pu se déplacer et qui ont suivi ma soutenance à distance.

J'ai eu la chance d'être entourée de personnes formidables durant cette thèse. Je me souviendrai toujours de ces années passées à IMT Atlantique et à l'ECAM. À toutes les personnes que je n'ai pas mentionnées ici, je tiens à vous dire combien je suis reconnaissante de vous avoir rencontrés. Vous avez rendu mon aventure mémorable. Merci infiniment à vous tous !

TABLE OF CONTENTS

List of Acronyms	12
List of Figures	15
List of Tables	18
Author’s List of Publications	21
Introduction	25
Context and background	25
Challenges faced by AIVs	26
Outline and research questions	28
Thesis organization	31
1 State-of-the-art	35
1.1 Introduction	35
1.2 Autonomous industrial vehicles	36
1.2.1 Integration of AIVs in Industry 4.0	36
1.2.2 Main functions performed by an AIV	40
1.2.3 Obstacle avoidance	42
1.2.4 Task allocation	48
1.2.5 Cooperative perception	52
1.2.6 Energy management	58
1.3 Collective mobile robot autonomy	62
1.3.1 Mobile robot cooperation	62
1.3.2 Collective intelligence	63
1.3.3 Collective autonomy	64
1.4 Methodologies and tools used to simulate and experiment of AIVs	68
1.4.1 Simulation in the field of AIVs	68
1.4.2 Agent-based systems	69

TABLE OF CONTENTS

1.4.3	Advanced decision-making techniques for AIVs	74
1.4.4	Fuzzy logic systems	75
1.4.5	Emulators and simulators to move towards real experiments in the field of AIVs	77
1.5	Conclusion	81
2	Collective Obstacle Avoidance Strategies	87
2.1	Introduction	87
2.2	Contribution of a collective strategy to an obstacle avoidance algorithm . .	88
2.2.1	Algorithm improvement	89
2.2.2	Agent-based model simulation	90
2.2.3	Environment simulation and scenarios	93
2.2.4	Simulation results	95
2.2.5	Discussion	97
2.3	Global obstacle avoidance by AIV with a collective strategy	99
2.3.1	Obstacle avoidance approaches: scenarios	100
2.3.2	Environment simulation and setting up the experiments	102
2.3.3	Experiments results	105
2.3.4	Discussion	107
2.4	Preparation of real experiments of an intersection scenario	108
2.4.1	Representation of ETSI messages for the industrial context	108
2.4.2	Experimentations context and settings	111
2.4.3	Communication	114
2.4.4	Results	115
2.4.5	Discussion	116
2.5	Conclusion	119
3	Collective Task Allocation Strategies	123
3.1	Introduction	123
3.2	Dynamic TA strategy in V2X cooperation mode with the infrastructure . .	124
3.2.1	Formal TA problem	125
3.2.2	Communication	126
3.2.3	Tasking process	126
3.3	Dynamic task (re-)allocation simulation	128
3.3.1	Agent-based model simulation for TA scenarios	129

3.3.2	Environment simulation	130
3.3.3	Scenarios	132
3.3.4	Results	135
3.3.5	Discussion on agent properties	139
3.4	Possible impact on TA: trust on different messages	141
3.4.1	Scenario	142
3.4.2	Results	143
3.5	Conclusion	143
4	Cooperative Perception Strategies	147
4.1	Introduction	147
4.2	Cooperative perception context	148
4.2.1	Input data	149
4.2.2	Cooperative map architecture analysis	150
4.2.3	Global architecture for V2X data fusion	151
4.3	Architecture proposed in Carla Simulator	154
4.3.1	Scenario considered	154
4.3.2	CAM: Self-localization of vehicle	155
4.3.3	Building CPM	157
4.4	Evaluation of the global architecture	159
4.4.1	Implementation of the architecture	159
4.4.2	Discussion	162
4.5	Conclusion	163
5	Collective Energy Management Strategies	167
5.1	Introduction	167
5.2	Fuzzy decision model for battery recharging	169
5.2.1	Fuzzy model	169
5.3	Fuzzy agent-based simulation for managing battery recharging	173
5.3.1	Case study: autonomous management of battery recharging	174
5.3.2	Comparisons between thresholds and fuzzy logic models	175
5.3.3	Increases in fuzzy logic criteria	178
5.4	Problem of recharging batteries with TA	180
5.4.1	Basic TA auction strategy with fuzzy rules	181
5.4.2	Energy management with fuzzy rules	183

TABLE OF CONTENTS

5.4.3	Improving the fuzzy recharge model	186
5.5	Conclusion	189
Conclusion		193
Contributions	194
Future works	197
A Symbols Table		203
B Multi-Agent simulation platforms		206
B.1	"4-Lace Circuit" simulation	206
B.2	ACT simulation	207
B.3	Industrial circuit simulation	208
B.4	Task allocation simulation	210
B.5	Airport baggage handling simulation	212
C Method for estimating AIV positions		215
C.1	Context	215
C.2	Localization AIV model	216
C.3	Mathematical position estimation model	217
D Methodology proposed for energy management		220
D.1	Context	220
D.2	Methodology with a realistic point of view	221
D.2.1	Optimal control problem	221
D.2.2	Energy control strategies	224
D.2.3	Fuzzy model for energy control	227
D.2.4	Numerical simulation in Matlab	228
E Fuzzy agent-based simulation		233
E.1	Agent-based fuzzy system	233
E.2	Behaviour of a fuzzy agent	233
E.3	Behavioural functions of a fuzzy agent	234
E.4	Fuzzy interactions between two fuzzy agents	234
Bibliography		235

Résumé long en français

255

LIST OF ACRONYMS

- ABMS** Agent-Based Modelling Simulation.
- ACK_MCM** Acknowledgment Maneuver Coordination Message.
- ACO** Ant Colony Optimization.
- ADAS** Advanced Driver Assistant System.
- AGV** Automated Guided Vehicle.
- AI** Artificial Intelligence.
- AIV** Autonomous Industrial Vehicle.
- AMR** Autonomous Mobile Robot.
- AV** Autonomous Vehicle (in the road domain).
- CACC** Cooperative Adaptive Cruise Control.
- CAM** Cooperative Awareness Message.
- CAV** Connected and Automated Vehicle.
- CD** Complex interDependencies.
- C-ITS** Cooperative-Intelligent Transport Systems.
- CPM** Cooperative Perception Message.
- CRM** Cooperative Response Message.
- CTM** Cooperative Task Message.
- DENM** Decentralized Environmental Notification Message.
- ETSI** European Telecommunications Standards Institute.
- GNSS** Global Navigation Satellite System.
- GPS** Global Positioning System.
- HMI** Human-Machine Interface.

ICT Information Communication Technologies.

ID Intra-robot interDependencies.

IMU Inertial Measurement Unit.

IOT Internet of Things.

IR Industrial Revolution.

ITS Intelligent Transport System.

LiDAR Light Detection And Ranging.

MAPEM MAP Extended Message.

MAS Multi-Agent System.

MCM Maneuver Coordination Message.

MR Multiple Robots.

MRTA Multi-Robot Task Allocation.

MRTD Multi-Robot Task Decomposition.

MT Multiple Tasks.

MTMR Multiple Tasks for Multiple Robots.

MTSR Multiple Tasks for a Single Robot.

ND No interDependencies.

OCP Optimal Control Problem.

PMP Pontryagin's Maximum Principle.

PSO Particle Swarm Optimization.

RADAR RAdio Detection And Ranging.

ROS2 Robot Operating System 2.

RRT Rapidly-exploring Random Trees.

RSU Road Side Unit.

RWW Road Works Warning.

SAE Society of Automotive Engineers.

SLAM Simultaneous Localization And Mapping.

SR Single Robot.

ST Single Task.

STSR Single Task for a Single Robot.

TA Task Allocation.

V2I Vehicle-To-Infrastructure.

V2P Vehicle-To-Pedestrian.

V2V Vehicle-To-Vehicle.

V2X Vehicle-To-Everything.

VRU Vulnerable Road User.

XD Inter-robot Dependencies.

LIST OF FIGURES

1.1	Nine pillars defined for Industry 4.0	37
1.2	AIV control functional diagram	41
1.3	Illustration of V2X communications	54
1.4	Navigation process inspired by [Van+18]	54
1.5	Agent properties	70
1.6	Rasmussen’s model of the operator: three levels of behaviour adapted from [Ras83]	71
2.1	Improvement of Bahnes’s algorithm to deal with the correlated problem of collision and obstacle	91
2.2	Agent-based model for AIVs obstacle avoidance	92
2.3	Simulation of a LiDAR use	94
2.4	Simulation of the Bahnes’ algorithm: a) on the right top side of picture two green and red AIVs arrive at an intersection, b) the green AIV passed the intersection after communicating with other AIVs, and the red AIV waits to cross the intersection	94
2.5	Bahnes’ algorithm for intersection crossing	95
2.6	Situation of intersection crossing by three AIVs	99
2.7	Cooperative messages used for intersection crossing	100
2.8	<i>IndustrialCircuit</i> diagram	103
2.9	Oriented graph of the <i>IndustrialCircuit</i>	103
2.10	Application interface to simulate missions in the warehouse environment	104
2.11	Gazebo Simulation environment for experimentations	113
2.12	Nodes and topics representation for <i>robot1</i> for path planning	113
2.13	Process of publishing/subscribing topics for <i>position_control</i> node of each robot	114
2.14	Nodes and topics representation for exchanging messages	115
3.1	V2X cooperation	128

LIST OF FIGURES

3.2 CTM and CRM exchanged during TA 129

3.3 Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple 130

3.4 Representation of the *IndustrialCircuit* for TA simulation scenarios . . . 131

3.5 Directed graph corresponding to the circuit 132

3.6 Multi-agent simulation interface 132

3.7 Scenario Sc_1 134

3.8 Scenario Sc_2 134

3.9 Scenario Sc_3 135

3.10 Scenario Sc_4 142

4.1 Cooperative perception in a warehouse 150

4.2 Representation of data before and after fusion 152

4.3 Global architecture for data fusion 153

4.4 Problematic scenarios: view of the camera infrastructure © IEEE 2024 . . 156

4.5 V2X messages in ROS2 topics generated from the scenario in **Figure 4.4** © IEEE 2024 158

4.6 Global architecture: Carla-OpenCDA-ROS2 © IEEE 2024 159

4.7 Sequence diagram of the communication and object detection process in the scenario depicted in **Figure 4.4b** involving two CAVs and one RSU © IEEE 2024 161

4.8 ROS2 communication flow of CAM and CPM in the scenario depicted in **Figure 4.4b** © IEEE 2024 162

5.1 Energy level 171

5.2 Distance to target 171

5.3 Distance to energy source 171

5.4 Decision 172

5.5 Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple. 174

5.6 Airport baggage handling simulation HMI 175

5.7 Oriented graph: distance in the environment in meters 176

B.1 "4-Lace Circuit" simulation interface 207

B.2	Simulation of radar use: a - at the top of the picture: one blue and three yellow AIVs arrive near the intersection. b - while waiting for the yellow AIV to pass the intersection, the radar of the blue AIV and the two other yellow AIV allow them to stop and keep their distance to avoid colliding.	208
B.3	Simulation of the scenarios: a - in the center of the picture: a blue and yellow AIV arrive at an intersection. b - the yellow AIV passed the intersection after communicating with other AIVs. c - on the left side of the picture: a red AIV perceives a fixed obstacle in front of him. d - a red AIV avoided the obstacle	209
B.4	Application interface to simulate missions in the warehouse environment	210
B.5	Multi-agent simulation interface	212
B.6	Airport baggage handling simulation HMI	213
B.7	Airport baggage handling simulation parameters	214
C.1	Agent-based model for AIVs obstacle avoidance and estimation positions	217
C.2	Algorithm for updating the position of an AIV agent	219
D.1	Example: optimal command for $T > T_{min}$ with $T_{min} = 9.79$ and speed over a distance 30 m	224
D.2	Map of Strategies in red color: oriented graph nodes	229
D.3	Position-Speed phase diagram: <i>Strategy1</i>	230
D.4	Energy consumed by the <i>Strategy1</i>	230
D.5	Map <i>Strategy1</i> : oriented graph nodes with their values distance (grey color) and speed (red color)	230
D.6	Position-Speed phase diagram: <i>Strategy2</i>	231
D.7	Energy consumed by the <i>Strategy2</i>	231
D.8	Map <i>Strategy2</i> : oriented graph nodes with their values distance (gray color) and speed (red color)	231
D.9	Position-Speed phase diagram: <i>Strategy3</i>	232
D.10	Energy consumed by the <i>Strategy3</i>	232
D.11	Map <i>Strategy3</i> : oriented graph nodes with their values distance (gray color) and speed (red color)	232

LIST OF TABLES

1.1	Taxonomy summary of TA problems in multi-robot systems	50
1.2	ITS messages used in the manuscript	56
1.3	ETSI Infrastructure-specific messages [ITS20]	57
1.4	V2X messages classified according to the four classes of cooperation	65
1.5	Type of agents [RAT19]	70
1.6	Type of learning agents [RND10]	72
1.7	Definitions for ABMS based on agent properties [Mac16]	73
1.8	The different categories based on multiple criteria of MAS [RAT19]	74
2.1	S_{c_1} : anti-collision algorithm simulation results	96
2.2	S_{c_2} : obstacles avoidance algorithm simulation results	97
2.3	S_{c_2} : anti-collision and obstacles avoidance algorithm cost	97
2.4	Correspondence between Bahnes' algorithm messages and ETSI messages	98
2.5	Completed missions in relation to the strategy number of agents	106
2.6	Total distance travelled by agents in relation to the strategy number	106
2.7	Total execution time to carry out the missions in relation to the strategy number	106
2.8	Representation of CAM Message for the industrial context	108
2.9	Modelization of CAM in ROS2	109
2.10	Representation of DENM for the industrial context	110
2.11	Modelization of DENM in ROS2	110
2.12	Representation of CPM for the industrial context	111
2.13	Modelization of CPM in ROS2	112
2.14	Representation of MCM for the industrial context	117
2.15	Modelization of MCM in ROS2	118
2.16	Representation of ACK_MCM for the industrial context	118
2.17	Modelization of ACK_MCM in ROS2	119
3.1	The various V2X communications in scenarios: S_{c_0} , S_{c_1} , S_{c_2} and S_{c_3}	136

3.2	Test sets for the scenarios	137
3.3	Obstacle obstructing an aisle, detected by the <i>AIV4</i> agent	137
3.4	Obstacle obstructing an aisle, detected by a camera agent	138
3.5	Breakdown during part 1 of the <i>AIV2</i> mission	139
3.6	Breakdown during part 2 of the <i>AIV2</i> mission	139
3.7	Stock point n°15 has become inaccessible	140
3.8	The stock point n°15 has become inaccessible, then cleared by a worker (human operator)	140
3.9	Agent properties captured in the scenarios	141
3.10	The various V2X communications in the scenario Sc_4	142
3.11	A worker checks the presence of camera-detected obstacle	143
5.1	Time results for 1000 bags for Sc_1 , Sc_2 and Sc_3	176
5.2	Performance indicators for Sc_1	177
5.3	Performance indicators for Sc_2	177
5.4	Performance indicators for Sc_3	178
5.5	Time results and configuration for 1000 bags for Sc_4 , Sc_5 and Sc_6	179
5.6	Performance indications for Sc_4	179
5.7	Performance indicators for Sc_5	180
5.8	Performance indicators for Sc_6	180
5.9	Global results of the strategy <i>FuzzyBasicTa</i>	184
5.10	Global results of the strategy <i>FuzzyBasicTaAndRecharge</i>	186
5.11	Results linked to recharges with the <i>FuzzyBasicTaAndRecharge</i> scenario	186
5.12	Global results of the strategy <i>FuzzyBasicTaAndImprovedRecharge</i>	189
5.13	Results linked to recharges with the <i>FuzzyBasicTaAndImprovedRecharge</i> scenario	189
A.1	Symbols table	204
A.2	Linguistic variables table	205

AUTHOR'S LIST OF PUBLICATIONS

Book chapter

- [ARAC24] J. Grosset, A.-J. Fougères, C. Couturier, M. Djoko-Kouam, and J.-M. Bonnin, « Collision and Obstacle Avoidance for Industrial Autonomous Vehicles – Simulation and Experimentation Based on a Cooperative Approach. », *in: Advances in Robotics and Automatic Control*, Sergey Y. Yurish, International Frequency Sensor Association (IFSA) Publishing, vol. 3, 2024, pp. 1–23, ISBN: 978-84-09-57872-6.

Journal articles

- [ALGO24] Juliette Grosset, Ouzna Oukacha, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Fuzzy Multi-Agent Simulation for Collective Energy Management of Autonomous Industrial Vehicle Fleets », *in: Algorithms* 17.11 (Nov. 2024), p. 484, ISSN: 1999-4893, DOI: 10.3390/a17110484.
- [ICAE24] J. Grosset, A.-J. Fougères, M. Djoko-Kouam, and J.-M. Bonnin, « Multi-Agent Simulation of Autonomous Industrial Vehicle Fleets: Towards Dynamic Task Allocation in V2X Cooperation Mode », *in: Integrated Computer-Aided Engineering* 31.3 (Apr. 2024), pp. 249–266, ISSN: 10692509, 18758835, DOI: 10.3233/ICA-240735.
- [ICAE23] J. Grosset, A. Ndao, A.-J. Fougères, M. Djoko-Kouam, C. Couturier, and J.-M. Bonnin, « A Cooperative Approach to Avoiding Obstacles and Collisions between Autonomous Industrial Vehicles in a Simulation Platform », *in: Integrated Computer-Aided Engineering* 30.1 (Jan. 2023), pp. 19–40, ISSN: 1069-2509, DOI: 10.3233/ICA-220694.

Work done prior to thesis

- [SENSORS23] D. Bouchabou, J. Grosset, S. M. Nguyen, C. Lohr, and X. Puig, « A Smart Home Digital Twin to Support the Recognition of Activities of Daily Living », *in: Sensors* 23.17 (Jan. 2023), p. 7586, ISSN: 1424-8220, DOI: 10.3390/s23177586.

International conference papers

- [VTC24] Juliette Grosset, Jean-Marie Bonnin, Alain-Jérôme Fougères, Manabu Tsukada, and Moïse Djoko-Kouam, « Generation of V2X Messages from Carla Simulator for Cooperative Perception: Application to Pedestrian Safety. », *in: 2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*, Washington D.C., USA, Oct. 2024.
- [AISYS24] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Fuzzy Agent-Based Simulation of Integrated Solutions for Task Allocation and Battery Charge Management for Fleets of Autonomous Industrial Vehicles », *in: AI-based Systems and Services (AISyS) 2024*, Venice, Italy, Oct. 2024.
- [ASPAI24] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Fuzzy Agent-Based Simulation for Managing Battery Recharging for a Fleet of Autonomous Industrial Vehicles », *in: ASPAI 2024: 6th International Conference on Advances in Signal Processing and Artificial Intelligence*, Funchal (Madeira), Portugal, Apr. 2024, **Best Paper Award**.
- [ASPAI22] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Collective Obstacle Avoidance Strategy - an Agent-Based Simulation Approach », *in: ASPAI 2022: 4th International Conference on Advances in Signal Processing and Artificial Intelligence*, Corfu, Greece, Oct. 2022.
- [ARCI22] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Christophe Couturier, and Jean-Marie Bonnin, « Simulation of a Collision and Obstacle Avoidance Algorithm for Cooperative Industrial Autonomous Vehicles », *in: 2nd IFSA Winter Conference on Automation, Robotics & Communications for Industry 4.0 (ARCI' 2022)*, Andorra la Vella, Andorra, Feb. 2022.

National communications

- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Talk on "Multi-Agent Simulation of Autonomous Industrial Vehicle Fleets. Collective Energy Management with V2X Cooperation". *Colloque IMT 2024: Industrie du Futur Responsable*, Mines Saint Etienne - Campus Aix-Marseille-Provence, April 23-25, 2024, **Best Presentation Award ex-aequo**.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Poster presentation on "Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles. Traffic hazards simulated by fuzzy agents". *32ème Rencontres Francophones sur la Logique Floue et ses Applications (LFA 2023)*, Bourges, November 9-10, 2023.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Talk presentation on "Using AI for Cooperative Perception". *Colloque IMT 2023: Data & AI - IMT Atlantique*, Nantes, November 8-9, 2023.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Poster presentation on "Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles. From simulation to real experiments". *Colloque IMT 2022: Les sciences de l'information au service des nouvelles mobilités*, Palaiseau, October 13, 2022, **Best Poster Award ex-aequo**.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Talk and Poster presentation on "Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles". *JFSMA 2022 - Sessions jeunes chercheurs aux Journées Françaises des Systèmes Multi-Agents - Ecole des Mines, Saint-Etienne*, 27 June to 1 July 2022.

Others

- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Talk on "Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles". *Jeudi de la Recherche 2024 ECAM Rennes, April 11, 2024*.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Jean-Marie Bonnin. Talk on "Collective obstacle avoidance strategy - an agent-based simulation approach". *Jeudi de la Recherche 2023, ECAM Rennes, March 16, 2023*.

- Anne-Gwenn Bosser, Liana Ermakova, Florence Dupin de Saint-Cyr, Pierre de Loor, Victor Charpenay, **et al.**. Poetic or Humorous Text Generation: Jam Event at PFIA2022, *in: 13th Conference and Labs of the Evaluation Forum (CLEF 2022)* Bologna (Italy), Sep 2022., pp. 1719-1726.
- **Juliette Grosset**, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Christophe Couturier, Jean-Marie Bonnin. Poster presentation on "Towards increasing the autonomy of Autonomous Industrial Vehicles using collective intelligence strategies". *Jeudi de la Recherche 2022 ECAM Rennes*, March 24, 2022.

INTRODUCTION

Context and background

The advent of Industry 4.0 has introduced new challenges in optimizing the flow of data, products, and materials within production environments. Autonomous Industrial Vehicles (AIVs), including Automated Guided Vehicles (AGVs) and other Autonomous Mobile Robots (AMRs), have emerged as promising solutions to address these challenges. However, the introduction and deployment of fleets of AIVs in companies remains problematic on a number of levels: acceptability to employees, precise vehicle localization, maintaining smooth traffic flow, and ensuring reliable perception in dynamic and ever-changing environments and so on. As a result, the autonomy of AIVs is constrained by predetermined trajectories and rigid operational frameworks.

This thesis focuses on enhancing **the autonomy of AIV fleets by leveraging collective intelligence strategies**. By enabling robust communication and exchange of relevant traffic information between AIVs and the infrastructure, we aim to improve their adaptability, decision-making capabilities, and overall efficiency. The key areas of interest include:

- Adaptation to circulation constraints: Improving AIVs’s capacity to adjust to changing environments, such as dynamic storage areas and production lines, to ensure seamless operation despite changing conditions. This adaptability leverages advancements in Artificial Intelligence (AI) and Internet of Things (IOT) technologies [Kha+20] for enhanced environmental perception.
- Improved decision-making: developing strategies that allow AIVs to make informed decisions despite incomplete, uncertain, or fragmented information [Mi21].
- Vehicle-To-Everything (V2X) communication: facilitating communication between AIVs within the fleet, with the infrastructure, and with human operators, thereby improving coordination and safety [Mi21].
- Energy efficiency: implementing strategies to reduce the energy consumption of AIV fleets, even under varying traffic and operational constraints [BM16].

Challenges faced by AIVs

AIVs are becoming increasingly integral to modern industrial settings, driving efficiency and automation in various applications. However, their integration and operation come with a range of challenges that must be addressed to fully realize their potential. These challenges can be broadly categorized into technical, operational, and human factors.

Technical Challenges

- **Navigation and localization:** AIVs must navigate complex and dynamic environments while maintaining accurate localization. The ability of AIVs to **perceive** their environment and avoid obstacles is fundamental. Indeed, **Obstacle avoidance** is a critical function for AIVs, enabling them to navigate safely, particularly in complex and dynamic environments with high traffic or frequent changes [RK22]. The reliability of sensors and the robustness of algorithms for real-time localization are critical for effective navigation. The challenge lies in developing reliable perception systems that can detect and differentiate between static and dynamic objects, even under adverse conditions like poor lighting. Additionally, real-time motion planning and collision avoidance algorithms must be robust enough to handle unexpected obstacles without causing operational delays [HIA22].
- **Cooperative perception:** Inspired by advancements in the automotive sector, cooperative perception is essential for enhancing situational awareness among AIVs in fleet operations. The challenge is to develop effective communication protocols that allow AIVs to share and integrate sensory information in real-time, especially in environments [Per+20]. This cooperation is vital for ensuring safe and efficient navigation and task execution.
- **Task allocation:** Task allocation is crucial for optimizing the efficiency of AIVs fleets. The challenge involves designing algorithms that can dynamically assign tasks based on real-time conditions, such as vehicle availability, task priority, and environmental factors. Ensuring that these algorithms are scalable and adaptable to varying operational requirements without causing significant downtime is a key concern [DVD20].
- **Energy management:** Energy management is a pivotal component in the operational success of AIVs, directly influencing their efficiency and autonomy. The chal-

lenge is to optimize energy consumption through efficient algorithms and advanced battery technologies while ensuring that AIVs can complete their tasks without frequent interruptions for recharging or battery replacements [BM16]. Balancing energy use with task demands remains a critical area of development.

Operational challenges

- **Integration with existing systems:** Integrating AIVs into existing industrial systems and workflows can be complex [And+15; BKD18]. It requires harmonizing AIV operations with legacy systems, adjusting production processes, and ensuring compatibility with current infrastructure. Effective integration strategies and adaptable system designs are necessary for successful deployment [PR17].
- **Communication and coordination:** AIVs need to communicate and coordinate effectively, especially when operating in fleets. Ensuring reliable communication between AIVs and with central systems is challenging, particularly in environments with heterogeneous fleet of AIVs. The development of robust communication protocols and decentralized decision-making strategies is crucial for seamless operation [Per+20].
- **Scalability and flexibility:** As industrial demands evolve, AIV systems must be scalable and flexible to accommodate varying operational requirements [DVD20]. Adapting AIVs to different tasks or production lines without significant downtime or reconfiguration presents a significant challenge. Modular and adaptable AIV designs can help address this issue.
- **Safety and reliability:** Ensuring the safety and reliability of AIVs is critical to prevent accidents and maintain operational continuity. This includes developing fail-safe mechanisms, ensuring compliance with safety standards, and implementing robust error detection and recovery systems. The cybersecurity of AIVs fleets has so far been little studied. The closest context is automotive cybersecurity. In context, the reference is *ISO/SAE 21434:2021*, which governs automotive cybersecurity with the aim of pave the way for connected vehicles and mitigate the cybersecurity risks posed to passenger vehicles. Advanced Driver Assistant System (ADAS) are tending to join the sphere of autonomous vehicles, but this standard nevertheless constitutes a relevant basis for establishing the cybersecurity of our AIV fleets. In addition, the use of AI techniques to infer a cyber threat level is a new approach that makes it possible to cover so-called 0-days attacks, i.e. attacks that have never

been implemented in the past.

Human factors

- **Acceptance and trust:** The acceptance and trust of AIVs by human operators and stakeholders are crucial for successful implementation. Resistance to change, concerns about job displacement, and skepticism about the reliability of AIVs can hinder their adoption.
- **Training and skill development:** Operators and maintenance personnel must be adequately trained to interact with and manage AIV systems. This includes understanding their operation, troubleshooting issues, and performing routine maintenance.
- **Cooperation with robots:** Co-bots, or collaborative robots, are designed to assist humans in tasks, improving productivity and reducing physical strain. However, challenges arise in ensuring smooth interaction, as it requires intuitive controls, clear communication, and strong safety protocols to avoid accidents and miscommunication.

Addressing these challenges requires a multidisciplinary approach, combining advances in technology with thoughtful consideration of operational and human factors. By overcoming these hurdles, the full potential of AIVs can be realized, leading to enhanced efficiency, safety, and flexibility in industrial environments.

Outline and research questions

The integration of AIVs into modern industrial systems presents a range of challenges, especially as we aim to enhance the autonomy of these vehicles within a fleet. Future autonomous robotic systems will likely consist of diverse robots working collaboratively to accomplish complex missions. The synergy between these heterogeneous robots and the potential to combine their unique capabilities could significantly benefit a multitude of diverse and extensive applications [RK22].

In this manuscript, we will not delve into human challenges related to AIVs or cybersecurity concerns. We made the strong assumption that communication systems are reliable and free from cyber threats. Instead, our primary focus is on the technical challenges involved in increasing AIV autonomy, particularly through the implementation of

collective strategies within a fleet. The main objective of this thesis is to propose and evaluate **collective intelligence strategies** that can significantly enhance the **autonomy and efficiency of AIVs** operating within industrial fleets. The following research questions form the foundation of this thesis:

How can the navigation and localization capabilities of AIVs be enhanced in complex, dynamic environments through cooperative perception and collective strategies?

This question explores how AIVs can be enabled to perceive and navigate their environments more accurately. We will investigate how sharing sensory information among AIVs can improve situational awareness and navigation precision. Additionally, we will examine how local processing can be optimized to achieve better global performance through collective strategies. The interaction with infrastructure and the shared local perception among mobile robots within a fleet are critical for improving overall fleet autonomy under strong cooperation hypotheses.

What strategies can optimize task allocation within a fleet of AIVs to enhance operational efficiency and reduce downtime?

This question addresses the challenge of dynamic task allocation. We will explore how collective strategies can be leveraged to develop scalable algorithms that efficiently distribute tasks based on real-time conditions and fleet availability, thereby enhancing operational efficiency.

How can energy management be optimized across a fleet of AIVs to extend operational autonomy and reduce interruptions due to recharging or battery replacement?

This question focuses on the challenge of energy management. We will examine collective energy management strategies that take into account both individual AIV energy levels and the operational demands of the entire fleet, aiming to extend operational autonomy and minimize disruptions.

How can the heterogeneity of industrial systems be managed more effectively?

This question seeks to demonstrate how Intelligent Transport System (ITS) technologies can be adapted to industrial contexts, enabling localized processing and control to improve overall performance from a global perspective.

Indeed, once the scenarios and communication technology constraints in terms of performance (e.g., delay, jitter, throughput, and availability) are identified, the selection of suitable technology will depend not only on technical characteristics but also on anticipated developments and industrial constraints. The ongoing debate and research into the comparative advantages of the main candidate technologies for V2X communications in road environments [Baz+19] highlight the importance of examining the evolution of these technologies [NCP19] and their adaptation to the specific needs of industrial environments. However, as mentioned below, we will not focus on the setup of communication technologies.

What methodologies can be used to rigorously assess and refine collective strategies aimed at increasing the autonomy of AIV fleets?

This question addresses the need for robust tools and methodologies to evaluate and optimize collective strategies in AIV fleets. Developing the autonomy of AIVs requires a relevant working method. Identifying reusable or adaptable algorithms to address the various challenges of increasing autonomy is only part of the solution. It is also essential to model, simulate, test, and experiment with the proposed solutions to ensure their effectiveness and practical viability. We will explore simulation environments and modeling techniques, such as multi-agent systems, can be used to test and refine these strategies under various conditions. Additionally, we will investigate the integration of centralized and decentralized software engineering approaches to support the development and validation of these collective strategies. This inquiry is essential for systematically breaking down the complex autonomy challenges faced by AIVs and ensuring that the proposed solutions are effective in dynamic, real-world environments.

Moreover, our methodology involves decomposing the autonomy problem into manageable technical tasks, taking into account the inherent complexity of AIVs that navigate, perceive, plan the different tasks, and manage energy in dynamic environments. Each of these technical challenges will be addressed individually in the various chapters comprising the contributions of this thesis. By systematically addressing these tasks and applying collective strategies, we aim to enhance the overall performance and autonomy of AIV fleets.

Thesis organization

The research presented in this manuscript centers around four key technical functions that enhance the autonomy of an AIV:

- Obstacle avoidance
- Task allocation
- Cooperative perception
- Energy management

Chapter 1: This chapter provides a comprehensive overview of the current state-of-the-art in AMRs within the context of Industry 4.0. The chapter begins by introducing the integration of AMRs in industrial settings and outlines the core functions they perform, such as obstacle avoidance, task allocation, cooperative perception, and energy management. It then explores various methodologies and tools used to simulate and enhance these functions, emphasizing the importance of improving both individual and collective autonomy. The chapter also reviews existing emulators and simulators, highlighting their role in bridging the gap between simulations and real-world applications.

Chapter 2: This chapter delves into the development of **collective obstacle avoidance strategies** for AIVs in complex industrial environments. It begins by introducing a collective strategy for collision detection [ARCI22; ICAE23], followed by a proposed solution for global obstacle avoidance using collaborative approaches among AIVs [ASPAI22]. The chapter is also linked to a method for estimating AIV positions within a closed industrial setting detailed in **appendix C**, and concludes with the emulation of an intersection scenario to test the proposed strategies, paving the way for future real-world validations [ARAC24].

Chapter 3: This chapter focuses on the development of **collective task allocation strategies** for AIVs within complex industrial settings. It begins by introducing a dynamic task allocation strategy that leverages V2X cooperation, highlighting the need for effective communication and coordination to facilitate collective task allocation processes. The chapter then applies these strategies to a practical warehouse scenario, simulating the performance of AIV fleets under dynamic conditions to verify the effectiveness of the proposed approaches. Finally, it addresses the challenges posed by faulty agents in the infrastructure, proposing a beginning of solution to detect and mitigate their impact,

thereby enhancing the robustness of collective task allocation strategies. This chapter is related to the contribution published in the ICAE journal [ICAE24].

Chapter 4: This chapter explores **cooperation perception strategies** for AIVs, emphasizing the importance of precise environmental perception through cooperative systems. It begins by discussing how integrating data from multiple sensors and V2X communications can enhance situational awareness and decision-making within AIV fleets. The chapter then focuses on acquiring and utilizing road data from Cooperative-Intelligent Transport Systems (C-ITS) to improve environmental understanding in dynamic industrial settings. Finally, it presents an architecture for generating V2X messages within the Carla simulator, aiming to enhance pedestrian safety and cooperative perception, particularly at intersections [VTC24].

Chapter 5: This chapter delves into the development and implementation of **collective energy management strategies** for AIVs, focusing on recharging processes to enhance overall fleet performance. The chapter begins by introducing a fuzzy agent-based simulation, and then presents a fuzzy decision model specifically designed for battery recharging [AISYS24; ALGO24; ASPAI24]. By applying fuzzy logic, we aim to address and manage the uncertainties associated with the recharging process. The fuzzy decision model is applied in a real-world scenario, simulating the autonomous management of battery recharging within an airport environment, where simulation scenarios will be based on a proposed charging/discharging model for an AIV battery. Firstly, we compare fuzzy logic models with threshold-based approaches [ASPAI24]. Then, to create a more realistic simulation framework, the AIV energy model is further refined and an optimal control model is established to improve the energy consumption by each AIV during its mission, is detailed in **appendix D**. This work is currently under review [ALGO24]. The chapter concludes by integrating energy management with task allocation, illustrating the interdependence of these two aspects to ensure continuous operations and minimize downtime [AISYS24].

Conclusion: This thesis is the logbook of a journey through the complex field of collective intelligence strategies for AIVs, with a focus on improving core functionalities to enhance both individual AIV autonomy and the overall performance of AIV fleets. Throughout this work, we explored numerous challenges and untapped opportunities, aiming to make AIVs more capable of operating independently while also ensuring that fleets can work together more efficiently and safely. It became clear that current methods

have limitations, particularly in fully harnessing collective intelligence to optimize these critical functions. This final chapter summarizes the key steps of this manuscript, highlighting how they open new perspectives for future exploration aimed at the ultimate goal we have pursued over the past three years: developing strategies that not only enhance the autonomy of individual AIVs but also enable them to operate as a cohesive, intelligent fleet within industrial environments.

STATE-OF-THE-ART

1.1 Introduction

As presented in the introduction of the manuscript, improving the autonomy of a fleet of mobile robots in Industry 4.0 endures many challenges. This chapter provides an overview and discussion of the current state-of-the-art in AIVs within the context of Industry 4.0. It will explore the various methodologies and tools employed for simulating and experimenting with algorithms to improve the different functions performed by an AIV.

We identified that the implementation and deployment of fleets of AIVs in companies remains problematic at various levels: the acceptability of the employees, the location of the vehicles, the fluidity of traffic, the perception by the vehicles of changing and therefore dynamic environments, etc. The AIV autonomy is therefore reduced to predetermined trajectories. These autonomy limitations motivated us to seek solutions to facilitate the cooperation of AIVs fleet, aiming to improve their collective intelligence, which in turn improves AIV's individual autonomy as well. This hypothesis is extensively developed and explained throughout this thesis. This chapter is organized as follows:

First, we introduce the integration of AIV in Industry 4.0, and we present the main functions performed by AIV in **section 1.2**. We then narrow the focus by describing several works exploring the improvement of each function: obstacle avoidance, task allocation, cooperative perception and energy management.

Afterwards, in **section 1.3**, we delve into the state-of-the-art of collective mobile robot autonomy. On the one hand, we present mobile robot cooperation and explain the concept of collective intelligence in the literature for mobile robots. On the other hand, we share our vision of collective autonomy for AIVs.

In **section 1.4**, we present some general methodologies and tools used to simulate and experiment in the field of robotics. We will provide beforehand background concepts related to simulation in this field, such as agent-based and fuzzy logic simulations, in

order to provide a robust understanding of the underlying principles. Finally, we will review and discuss the potentials and limitations of the existing emulators and simulators. These tools aim to closely replicate real-world conditions, which make the transition to real experiments easier. Emphasis is placed on areas including network communication, traffic simulation, hybrid simulators, and the utilization of realistic simulators for training Autonomous Vehicles (AVs).

1.2 Autonomous industrial vehicles

1.2.1 Integration of AIVs in Industry 4.0

"**Industry 4.0** can be defined as a digital transformation making autonomous decentralized decisions in all cyber physical systems where each element works in interaction." [BKD19]

Industry 4.0 is the "fourth Industrial Revolution (IR)" builds upon the advancements of previous eras:

- 1st IR: steam-powered mechanization
- 2nd IR: mass production and electricity
- 3rd IR: electronics and Information Communication Technologies (ICT)

The 4th IR emerged in 2011 in Germany [KHW13]. Industry 4.0 leads to a strong digitalisation of industrial processes [Lu17; Xu+21b]. Boston Consulting Group (BCG) proposed a description of the future vision of Industry 4.0 with nine pillars of technological advancements shown in **Figure 1.1** [Rüß+15].

The notion of the factory becoming smarter through Industry 4.0 is characterized by continuous communication between the various tools and workstations integrated into production, storage and supply chains. The concept of Industry 4.0 has also been extended to other types of structures and sectors [TBS18], such as Supply Chain 4.0 [PGS19] or Airport 4.0 [TM21; ZP18].

In these contexts, the challenges revolve around enhancing the development and optimisation of data, product and material flow within production companies. Optimizing data and product flows involves several key aspects. Firstly, it includes integrating data from various sources to provide a comprehensive operational perspective. Secondly, it requires ensuring efficient communication of this data across different stages of the production process and among diverse systems and machines involved. Finally, optimizing

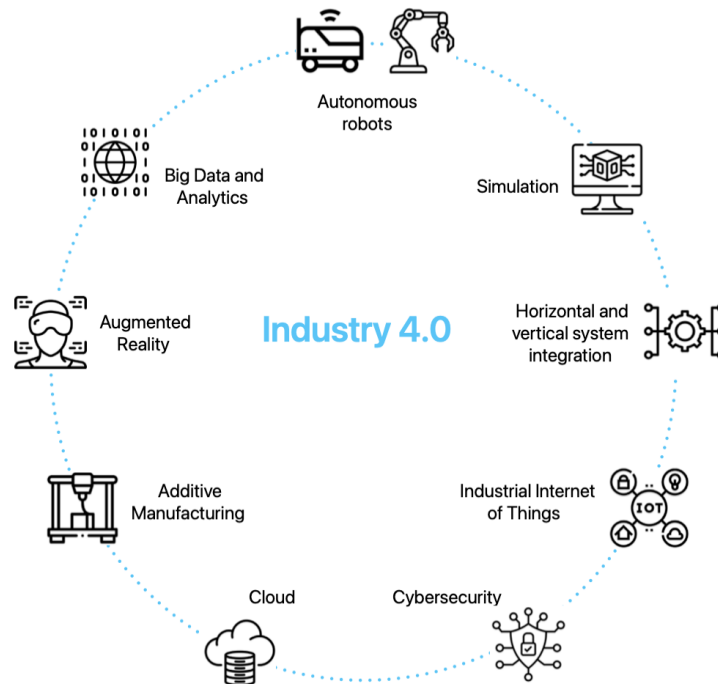


Figure 1.1 – Nine pillars defined for Industry 4.0 inspired by [Rüß+15]

product flows entails minimizing waste, reducing lead times, and maximizing throughput to enhance overall productivity and efficiency.

Achieving these objectives requires a significant increase in communication and cooperation among the component machines. Certain technological building blocks have been defined [PR17], notably for the use of autonomous systems [BKD18] and AGVs [And+15] or other autonomous and/or intelligent mobile robots. The first driverless vehicle was conceived in 1954 by Barrett Electronics Corporation, Illinois, a manufacturing company in the United States [Hy11]. It was renamed AGV in the 1980s [CCW19].

They are multiplying in factories, often in the form of fleets of vehicles, and their intelligence and autonomy are increasing. We will refer to these mobile robots designed to operate autonomously within industrial settings as Autonomous Industrial Vehicles (AIVs) throughout this manuscript.

[TBS18] defined the different key factors for integrating AIVs into a supply chain operations framework:

- **Physical characteristics**: allocation of space, arrangement of equipment and products, paths, aisles, width/height, floor type, lighting conditions.
- **Environmental conditions**: preservation conditions of materials and equipment,

health, safety and ergonomics for workers.

- **Operational characteristics:** demand, flow of people/equipment/materials, AIV characteristics, bin/pallet size/weight, scheduling, planning.
- **Equipment’s Hardware requirements:** type, material handling, coordination, communication, interoperability, compatibility.
- **Software requirements of the equipment:** interface type, coordination, communication, interoperability, compatibility.

In research on Intelligent Transport System (ITS), mainly in the context of smart cities [CAF17], the autonomy of autonomous vehicles has been well characterized with six autonomous driving levels distinguished by the Society of Automotive Engineers (SAE) [Wis22]. However, this is not the case for AIVs, and too little research exists in this area [EBW16]. In concrete terms, these levels mean:

1. **No driving automation:** Feet-on/Hands-on/Eyes-on/Brain-on.
2. **Driver assistance:** Feet-OFF/Hands-on/Eyes-on/Brain-on.
3. **Partial driving assistance:** Feet-OFF/Hands-OFF/Eyes-on/Brain-on.
4. **Conditional driving automation:** Feet-OFF/Hands-OFF/Eyes-OFF/Brain-on.
5. **High driving automation:** Feet-OFF/Hands-OFF/Eyes-OFF/Brain-OFF when the vehicle is within a specific area.
6. **Full driving automation:** Feet-OFF/Hands-OFF/Eyes-OFF/Brain-OFF and the vehicle can go anywhere.

The establishment and deployment of AIV fleets in an industrial context raises several challenges: the location of vehicles, the fluidity of traffic, the collision detection, the perception of disturbances in the environment (dynamics), the vehicle heterogeneity, the acceptability by employees and even their cooperation.

Currently, to achieve their tasks, AIVs, have navigation autonomy linked to rails, physical or virtual beacons. They are limited to predetermined trajectories and rely on the visibility or knowledge of each AIV position to avoid collisions. Their decision-making capacities are often limited to following these predetermined trajectories and stopping in the event of obstacles. Upon incidents, such as obstacles or breakdowns, route modifications are managed by the central system or a supervisor, who issues new mission orders to the vehicles.

The autonomy of AIVs is often limited to local visibility of their environment, as well as visibility and knowledge of other AIVs operating in the same environment. The ability of AIVs in the same fleet to exchange information between themselves, with the active elements of the infrastructure that they encounter on their route, or with the human beings who operate in their environment (faculty commonly referred to as V2X communications) [Per+20], should improve this autonomy in terms of:

- adapting to traffic constraints, particularly when the environment of a AIV changes over time (in the dynamic environments of baggage storage or routing areas, for example); ability to adapt, taking full advantage of the development of AI and IoT technologies [Mí21] to perceive environments;
- decision-making, even when the information available to an AIV is incomplete, uncertain or available but fragmented [Mí21];
- and reducing (or simply controlling) the energy impact, whatever the traffic constraints [BM16].

To increase the autonomy of an AIV, and even more the autonomy of a fleet of AIVs in a decentralized approach [DVS20], one way is to develop a collective intelligence to make the behaviours of vehicles adaptive. In this thesis, we aim to demonstrate that the decision-making autonomy of these vehicles, defined collectively, can help enhance the robustness and adaptability to changing traffic conditions. Thereby improving efficiency and safety in industrial environments.

The context of Factory 4.0 is increasingly leading to decentralized solutions, as centralization is showing its limits. Many authors have discussed the centralization or decentralization of AIV intelligence for the control of their activity, which is essential for their autonomy [Dra+16].

However, driven by future requirements such as flexibility, robustness and scalability, the current trend in AIV systems is towards decentralization. Decentralization, in this context, involves distributing the total intelligence of the system across its components: each device is allocated a portion of the total intelligence so that it can operate independently, striving to achieve the same overall objectives [DVD20]. By distributing intelligence across multiple machines, systems become more adaptable and resilient. They can respond more effectively to dynamic environments and unforeseen challenges without requiring constant oversight from a central authority or controller.

To optimize the management of AIVs in a factory or airport, and even more so to increase the autonomy of a fleet of AIVs in a decentralized approach [DVD20], one way is

to develop a kind of collective intelligence to make vehicle behaviour adaptive. However, before proposing collective optimisation of AIVs, it depends on the actual level of autonomy of the AIVs. Indeed, the initial step towards cooperation is therefore to increase the individual level of autonomy of each AIV.

1.2.2 Main functions performed by an AIV

What is the capabilities of an autonomous robot?

"A robot that is completely autonomous is able to observe its surroundings, make judgments based on what it sees and/or has been trained to recognize, and then carry out an action or manipulation in that environment." [LA23]

An AIV possesses the capability to autonomously navigate its surroundings if it can manage and control a set of fundamental tasks. These tasks, extensively discussed in existing literature, encompass perception, localization, mapping, task allocation, path planning, motion planning, and vehicle management [AA21; Per+20; RK22].

To achieve the autonomy of a mobile robot, [SNS11] subdivided it into four main sub-tasks:

- **Perception:** involves sensing and understanding the environment.
- **Localization:** determines the robot's position in the environment.
- **Path planning:** enables the robot to select a collision-free route to its destination.
- **Motion control:** governs its movements along this path.

Further elaboration on the functional schema of autonomous navigation was provided by [And+15], outlining requirements for systems comprising a fleet of AGVs in the *Safe AUtonomous NAvigation project*¹ (SAUNA). This schema includes navigation, task allocation, path planning, motion planning, and vehicle management.

We summarise the AIV control functional diagram in **Figure 1.2**, inspired by the SAUNA functional schema. Expanding on AGV system control modes, [DVD20] defined the total AGV-system control in centralized mode, decentralized mode, or hybrid mode, emphasizing components such as:

- **Navigation**
- **Task allocation**

1. SAUNA project: aass.oru.se/Research/mro/sauna

- Path planning
- Motion planning
- Vehicle management

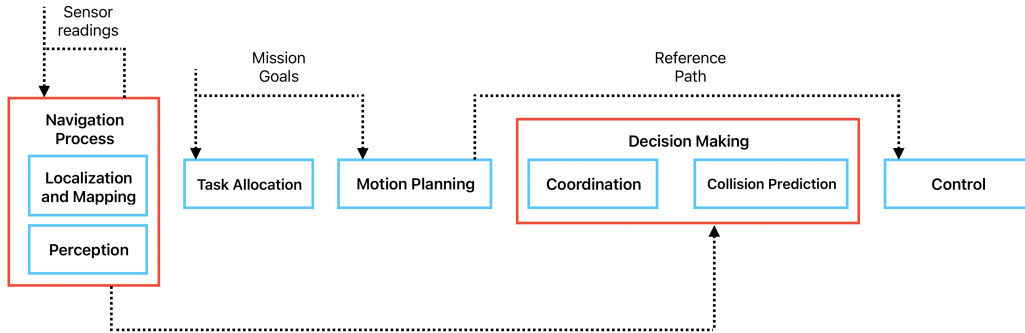


Figure 1.2 – AIV control functional diagram inspired by the SAUNA functional schema [And+15]

For autonomous navigation in dynamic environments, an AIV must simultaneously plan, locate, and map its surroundings [HIA22]. This involves four main components: controllers (microprocessors), actuators (motors), sensors (for obstacle avoidance, estimation, position, etc.), and power systems (components supplying energy to the robots, such as batteries) [RK22]. In current literature, perception and localization are included in the navigation process. Moreover, considerable research attention is increasingly directed towards task allocation and vehicle management, reflecting their growing significance in deploying larger fleets.

In the following subsections of this state-of-the-art on AIVs, we will delve into the latest advancements in obstacle avoidance during navigation, task allocation, perception, and energy management within the realm of autonomous navigation systems. Through these comprehensive analyses, we aim to provide insights into the current state of the art and outline future directions for advancing collective autonomy for AIV.

1.2.3 Obstacle avoidance

Relationship between obstacle avoidance and navigation

"The aim of navigation is to search an optimal or suboptimal path from the start point to the goal point with obstacle avoidance competence."
[Pan17]

Among the problems to be solved to make AIVs more autonomous, we can particularly identify the following:

1. **Avoidance of Collisions:** that prevents collisions between vehicles or with static/dynamic obstacles [HDA17; Nas+19];
2. **Path Planning or Path Finding:** that addresses obstacles obstructing predefined paths or hindering AIV passage [KHJ18; Kun+06; MK16].

Currently, avoidance strategies are often implemented on a per-robot basis [Kun+06], without a coordinated collective approach. However, the problem of avoidance is an individual problem that can be solved with a broader collective approach. By adopting a collective strategy, we can enhance the efficiency and effectiveness of obstacle avoidance. Addressing avoidance entails a multi-phase process encompassing:

- obstacle perception/detection,
- rerouting or trajectory planning (avoidance),
- and the overall strategy decisions.

In this manuscript, we will focus on specific aspects of obstacle avoidance while making certain strong assumptions:

1. **Obstacle perception/detection:** For the purpose of this discussion, we assume that the robot is already equipped with effective sensors and algorithms for detecting obstacles.
2. **Rerouting/trajectory planning:** We will not focus on specific rerouting algorithms, but we will use one of them in our experiments.
3. **Overall strategy decisions:** We will explore the broader strategic decisions that influence how obstacles are avoided, including:
 - Integration of multi-robot communication and coordination.
 - Adaptation to both static and dynamic obstacles.
 - Real-time decision-making processes and algorithms.

Efficient obstacle avoidance is paramount for enhancing the autonomy and safety of AIVs. To address this, a comprehensive approach is essential, encompassing key components such as localization, collision avoidance, path planning, and motion planning strategies. Here, we delve into each aspect, discussing their significance and the methodologies employed to tackle obstacles effectively. This includes considering both static and dynamic obstacles. Indeed, the solutions for obstacle avoidance depend on the nature of the obstacle, whether it is static or dynamic.

By addressing these aspects, we aim to show that moving towards a more collective and coordinated approach can significantly enhance obstacle avoidance capabilities, leading to improved autonomy and safety for AIVs.

Mapping and localization

SPLAM: Simultaneous Planning, Localization and Mapping

"A mobile robot should perform localization, mapping, and planning consecutively to operate successfully in the environment. If any of these three activities is absent, then a robot cannot walk autonomously in real-life deployment scenarios." [RK22]

Mapping is essential for path planning and obstacle avoidance. It involves creating a detailed map of the environment, which allows the robot to detect and respond to changes during navigation. This map can be pre-existing or generated in real-time [RK22].

Vehicles must therefore be given other means to locate themselves. Moreover, accurate localization is fundamental for AIVs to perceive and navigate their surroundings effectively. Localization consists of obtaining the exact location of the AIV on a map [Bar+03; CG15]. A position estimation provides an approximation of a vehicle's location to its environment; whether the environment is outdoor or indoor [Sal+99]. The literature on estimation theory is vast, encompassing a wide variety of techniques and ideas.

Naturally, the most common techniques receive frequent attention. These general techniques can be applied to a variety of problems, an example being parametric estimation methods such as weighted least squares estimators, maximum-likelihood estimators and minimum mean-square error estimators [Cox89].

Localization methods include:

- **Incremental or relative localization** [Ayn+17]: makes it possible to determine the position and orientation of a vehicle by taking into account its successive move-

ments from a known starting point.

- **Absolute localization** [Bar+03]: determines the vehicle’s position or robot in its external or internal environment using exteroceptive sensors. Absolute localization by definition avoids the drift over time that characterizes relative localization; the main disadvantage of this strategy is the loss of visibility of the landmarks in the environment that a vehicle uses to determine its position.

Therefore, two strategies are used for localization that rely on either natural or artificial landmarks (e.g., Global Positioning System (GPS) or beacons), respectively. Several types of techniques have been proposed in the literature: GPS, laser, natural (using Light Detection And Ranging (LiDAR)), sensor fusion, and vision-guided localization [Cam+20]. This localization can involve a physical path or a virtual path.

Two advanced localization techniques that integrate various sensors and algorithms are:

- **Map matching**: This technique aligns the vehicle’s estimated position with a known map, improving accuracy by correlating sensor data with mapped features. It helps refine the vehicle’s location estimate by matching its sensor readings with the known positions of landmarks on a pre-existing map.
- **Simultaneous Localization And Mapping (SLAM)** [TX21]: This technique involves creating a map of an unknown environment while simultaneously keeping track of the vehicle’s location within that map. Accurate localization is needed to build the map, and an accurate map is essential for precise localization.

These advanced techniques enhance the reliability and precision of localization by leveraging multiple data sources and sophisticated algorithms to mitigate the limitations of individual sensors.

Motion planning

Motion planning specifically addresses the vehicle’s dynamics, ensuring that the vehicle can navigate safely and efficiently while accounting for its physical and kinematic constraints. This process involves establishing/modifying the planned trajectory in real-time as a function of static or dynamic obstacles [Car+17]. The subtasks associated with this task are presented below:

- Centralised or decentralized collision avoidance: Techniques that focus on preventing collisions by either coordinating vehicle movements through a centralized system or allowing each vehicle to make independent decisions in a decentralized

manner, as explained in the subsection below 1.2.3.

- Deadlock avoidance: Strategies to prevent vehicles from becoming stuck due to conflicting movements. This requires understanding the vehicle's dynamic capabilities to ensure it can maneuver out of potential deadlock situations. A typical example of a deadlock situation is when vehicles attempt to cross an intersection simultaneously.
- Zone control: When the environment is divided into zones, each zone may have specific rules or controls that take into account the vehicle's dynamic properties. Zone control ensures that vehicles transition smoothly between different areas without causing congestion or collisions.

Different approaches have also been proposed to solve this problem: fuzzy logic, neural networks, and genetic algorithms, including hybrid approaches that are often more robust and give the best results [Pan17].

Obstacle perception and detection

An obstacle detection system

"A sophisticated obstacle detection system will detect any static and moving (dynamic) object in any driving environment and alarm for any potential risk of accidents and collisions." [YM20]

[YM20] conducted a review of recent technological advancements in obstacle detection. The study discusses various sensors:

- RAdio Detection And Ranging (RADAR): Effective for medium to long-range mapping, making it suitable for exploration, but it lacks the fine resolution needed for precise object identification.
- LiDAR: Offers high-resolution mapping capabilities for both short and long ranges, with varying degrees of coverage, including some systems that provide 360-degree views. Distance detection is one of the major functions of LiDAR. However, its performance is limited in challenging environments and with targets that have low reflectivity.
- Ultrasound: Cost-effective and performs well in short-range measurements.
- Camera (vision sensors): Capable of providing comprehensive environmental imagery across a wide range of conditions. It has a better performance in object detection and is only able to measure distance in a stereo version [Zie+16].

The exteroceptive sensors mentioned above are able to predict possible dangers. [Zie+16] conducted a survey of the capabilities of contemporary exteroceptive sensors and based on them Advanced Driver Assistant System (ADAS) system (discussed in **subsection 1.2.5**).

Combining various sensors with different operating principles, such as RADAR, infrared, and cameras, is a viable solution for obstacle perception and detection. Advances in hardware technology have led to performance improvements and cost reductions in LiDAR and depth cameras, enabling more accurate positioning and movement in robots [Che+21]. Additionally, multi-function LiDAR sensors, which integrate infrared and camera capabilities, offer similar applications to traditional LiDAR and camera sensors but with a reduced range [Zie+16].

However, sensors used in object detection and position estimation, such as RADAR, LiDAR and cameras [Yeo+21] are not perfect sensors [Mar+19]. In particular, they may be limited by their intrinsic properties, for example, camera object detection will be limited by the quality of the image as well as the distance between the sensor and the object [Hao+23]. Also, the process of calibration between the sensor intrinsic frame and the real-world map can induce some unprecise results independent of the sensors themselves [Jia+]. In addition, sensors are subject to failures [Saf+21] and cyberattacks [Wan+22] that could greatly impact the quality of the information they provide.

In this manuscript, we will characterise the different types of obstacles detected by AIVs:

- an obstacle that only slows down the AIV but which can be avoided without changing path, of varying size, defined as size s_1 or s_2 .
- an obstacle which cannot be avoided by the AIV, and which forces the AIV to reroute. In other words, the AIV must turn around and find another path, of greater or lesser size, defined as s_3 or s_4 .

Collision avoidance

Collision avoidance for AIVs requires a comprehensive approach to ensure safety and efficiency in dynamic environments. In this thesis, the term "*collision avoidance*" is used to describe a systematic process that includes detecting potential obstacles and rerouting trajectories if necessary to avoid collisions. Additionally, it consists also of relaying information through communication as well. By integrating these steps, AIVs can navigate in complex industrial environments while minimising the risk of accidents.

In the following, we will deal with avoidance through the choice of the route, the path but not the trajectory. These challenges manifest when vehicles confront scenarios like simultaneous intersection crossings, known as deadlock situations, or encounter obstacles obstructing their paths.

To address collision avoidance challenges, strategies are typically classified into centralized and decentralized approaches [Rai+22]. In a centralized collision avoidance framework, a central controller aggregates and processes data from all vehicles, coordinating their movements to prevent collisions. This method offers the advantage of optimizing traffic flow and decision-making efficiency but depends heavily on a robust communication infrastructure and the central controller's capacity to handle extensive real-time data. Centralized approaches are effective for managing small groups of robots; however, they become less practical for larger groups due to increased computational requirements.

In contrast, the decentralized approach is more effective, as it is less expensive computationally [Rai+22]. Indeed, decentralized collision avoidance enables each vehicle to independently make decisions based on locally available data and to communicate with neighbouring vehicles regarding their positions, speeds, and intended trajectories. This approach enhances scalability and reduces reliance on a single point of failure, but it necessitates the implementation of sophisticated algorithms to facilitate effective information exchange and coordination among vehicles without the need for a central authority.

Therefore, obstacle avoidance can be solved by enhanced cooperation [BKH16; Haf+13], i.e., the ability of AIVs to exchange relevant information with each other on their traffic conditions.

The collision avoidance algorithm proposed by [BKH16] makes it possible to deal with the priority of different vehicles when approaching an intersection. The solution requires the vehicle to know its own position and to be able to communicate with the other vehicles. It allows AIVs to communicate and cooperate using different types of messages. The communication between AIVs is done with 3 different types of messages:

- **Hello_msg**: message to indicate its presence with its position.
- **Coop_msg**: message before an intersection area to determine priority.
- **Ack_msg**: message to confirm receipt of a **Coop_msg**.

However, this algorithm does not deal with the problems of detection, communication and avoidance of fixed or moving obstacles (e.g. human operators). Indeed, if there is an obstacle or a problem in the environment, therefore the initial path is not possible to be done. Thus, we will propose an extension to the Bannes' algorithm in **chapter 2** -

section 2.2 [ARCI22; ICAE23], to effectively manage this communication following the detection of these different types of obstacles.

Moreover, solutions are needed to allow missions to be completed, and not be compromised because the main route defined by an AIV is obstructed. For that, path planning or path-finding algorithms [Kar+21; MK16] make it possible to re-plan its route, depending on whether the target of the mission has been reached or not.

Path planning

Indeed, Path planning consists of generating an obstacle-free path from point A, called the source, to point B, called the goal [KHJ18]. It involves [Pan17]:

- **Global path planning**: establishes routes from source to goal points.
- **Local path planning**: resolves displacements or trajectories following the presence of obstacles on the path.

Various types of algorithms and classical or heuristic-based methods have been proposed to solve the path planning problem: Dijkstra, A* and D* algorithms, Rapidly-exploring Random Trees (RRT), genetic algorithms, swarm algorithms (schools of fish, flocks of birds and colonies of ants), among others [HIA22]. The analysis of these different approaches and their comparisons are well documented in the literature and enable them to be classified according to their ability to respond to static or dynamic constraints in the environment [Kar+21].

1.2.4 Task allocation

Task Allocation (TA) consists of optimally assigning a set of tasks to be performed by agents, actors, robots or processes, grouped and organized within a cohesive system [GM04]. This is the case for mobile multi-robot systems [JM13; KHE15] or the AIV fleets addressed in this article [Kar+21]. Another pertinent example is the application in airports as presented by [ESI17].

The core objective of TA is to optimize resource utilization and overall system performance. In this sub-section, we will introduce a taxonomy of mobile robots TA, and explore various objectives associated with TA. Finally, we will review the state of the art in task planning for multiple robots, and decentralized task solutions.

Taxonomy

In the field of mobile robotics, the taxonomy presented in [DVD20] has been defined to better characterise this allocation and assignment functions:

- **Single Task for a Single Robot (STSR)**
- **Multiple Tasks for a Single Robot (MTSR)**
- **Multiple Tasks for Multiple Robots (MTMR)**

These classifications enable tasks to be assigned to one or multiple robots, with various tasks being allocated to heterogeneous or multitasking robots.

Moreover, [DVD20] defined also: allocation modalities such as:

- **instantaneous assignment**: In this approach, tasks are treated as independent, with no need for future planning since decisions are made based on the currently available information.
- **time-extended assignment**: This method considers interdependent tasks, allowing for the planning of future assignments while accounting for constraints such as synchronization, precedence, or time windows.

Numerous surveys, such as [KHE15], have exhaustively detailed the many combinations of these modalities in the context of Multi-Robot Task Allocation (MRTA).

Additionally, another critical dimension in classifying TA problems and assessing their solvability is the nature of interdependencies between tasks. This study [KSD13] identified the different levels of interdependencies:

- **No interDependencies (ND)**: Task execution values are independent of other tasks.
- **Intra-robot interDependencies (ID)**: Task execution values depend on the order in which they are executed by the assigned robot.
- **Inter-robot Dependencies (XD)**: Task execution values depend on the order in which tasks are executed by the multi-robot system.
- **Complex interDependencies (CD)**: Task execution values depend on their decomposition into sub-tasks.

We summarise this taxonomy to provide a comprehensive overview of the TA problem in **Table 1.1**.

Type of tasks	Type of robots	Type of allocation	Degree of interdependence
Single Task (ST)	Single Robot (SR)	instantaneous	None
			Intra-Robot
Multiple Tasks (MT)	Multiple Robots (MR)	extendeous	Inter-Robot
			Complex

Table 1.1 – Taxonomy summary of TA problems in multi-robot systems

Solution models and objectives

Different solution models have been proposed for TA:

- solutions based on **optimisation**: exact algorithms, dynamic programming, heuristics and meta-heuristics [KHE15];
- solutions based on **the contract net protocol**: particularly in the field of multi-agent systems (an initiating agent sends a call for proposals to the entire community of agents, chooses the best proposal received, then informs all the agent’s choice) [Woo02];
- solutions based on **the concept of the market**: announcement by an auctioneer, submission by bidders, selection by the auctioneer and award by the auctioneer [Dia+06].

Different types of optimisation objectives can be defined for this TA [DVD20]:

- **Cost objectives**: cost required for a robot to perform a task; this can be a travel cost such as time, distance or energy consumption.
- **Behaviour objectives**: the ability of a robot to perform a task.
- **Reward objectives**: payoff for performing a task.
- **Priority objectives**: urgency to perform a task.
- **Utility objectives**: subtracting the cost from the reward or fitness.

Multi-robot task planning

TA in multi-robot systems is complex and the tasks themselves may have many time, precedence or resource constraints [GM04].

It is then necessary to determine the objectives to be optimized, in particular among the travel cost (time, distance, fuel or battery consumption), the fitness (quality of task performing), the reward (gain of task completing), the priority (urgency of task completing), and the utility (balance between cost and fitness-reward) [DVD20].

In a global way, multi-robot task planning includes two processes:

1. the **Multi-Robot Task Decomposition (MRTD)**: refers to how a team mission can be decomposed into several subtasks which can be completed by the robots;
2. the **MRTA**: involves deciding how each subtask can be assigned to one robot [YJC13].

Thus, two major objective functions are defined for analysing a solution to the MRTA problem [Ste+19]:

- the makespan: the total time required to complete all assigned tasks, measured from the start time to the completion time of the last task. Mathematically, if we denote the completion time of all assigned tasks for the robot i as $completionTime_i$, then the makespan $robotTasksCompletionTime$ can be expressed as in **Equation (1.1)**. In this equation, n is the number of robots. This objective function is crucial in scenarios because minimizing the total operation time can significantly enhance efficiency.
- the sum of costs: the total time expended by all robots to reach their respective tasks. If $completionTime_i$ represents the time taken by robot i to complete its assigned tasks, then the sum of costs $totalCompletionTime$ is given by **Equation (1.2)**. In this equation, n is the number of robots. This objective function is crucial for minimizing individual robot time or energy use, such as in reducing overall energy consumption or balancing workload among robots.

$$robotTasksCompletionTime = \max_{i \in \{1, 2, \dots, n\}} completionTime_i \quad (1.1)$$

$$totalCompletionTime = \sum_{i=1}^n completionTime_i \quad (1.2)$$

Decentralized TA solutions

TA and planning are often managed centrally, even semi-centrally when global and local planning are differentiated [Car+17]. For the proper functioning of autonomous and dynamic systems, the requirements of flexibility, robustness and scalability, lead to consider decentralized mechanisms to react to unexpected situations.

Autonomy and decentralization are closely related concepts to the extent that an autonomous system operates and makes decisions independently [FAS20; TH19]. Additionally, a system is considered decentralized if the decision-making process is not controlled by a central authority.

The problem of TA [JM13; KHE15], tasks which grouped can constitute missions, must therefore be thought of in a decentralized way [DVS20].

For reasons of flexibility, robustness and scalability necessary in an Industry 4.0 or Airport 4.0 context, we are interested in decentralized TA solutions. These solutions, decomposed in this section below, must be able to assign tasks to a fleet of robots.

Particularly, solutions based on the market concept can easily be applied in a distributed context, where each mobile robot can become an auctioneer [Dao+21]. For each situation, a single mobile robot is appointed auctioneer [HK13]. He retains this role until the situation is definitively managed.

1.2.5 Cooperative perception

The advancements and solutions proposed in the road field for cooperative perception can be of great interest in the industrial sector. This section provides an overview of some of the work in this domain to highlight existing solutions.

As discussed, we saw in **subsection 1.2.2**, that perception and localization are included in the navigation process. The autonomous navigation process is a high-frequency recursive process. Perception involves the vehicle’s ability to sense and comprehend its surroundings through various sensors, including cameras, LiDAR, RADAR, and ultrasonic sensors.

SLAM, as mentioned in **subsection 1.2.3**, plays a critical role in this context by enabling the vehicle to construct a map of an unknown environment while simultaneously keeping track of its location within that environment. This is achieved through the integration of sensor data to update the map and refine the vehicle’s position, which is essential for accurate navigation and obstacle avoidance.

Currently, both AVs (in the road domain) and AIVs, face limitations in perception due to sensor costs and capabilities. For example, in the road context, [Van+18] recalled that most commercial vehicles offer only basic autonomy levels (1-2), requiring constant human supervision. Features like emergency braking and lane-keeping are common but full autonomy (Level 3) is limited to select models like Tesla’s Model S and Model X.

Moreover, every year, numerous car accidents occur globally, with 20% involving pedestrians. Ensuring road safety for everyone has become a crucial concern in the automotive industry. The European Union (EU) aims to achieve ”Vision Zero”, moving towards zero fatalities in road transport by 2050 [EU20].

Road vehicles must navigate through dense and diverse traffic, including Vulnerable

Road Users (VRUs), and their visibility is often constrained. Indeed, in certain situations, direct observation between vehicles and pedestrians may be impossible. This limitation can stem from factors such as the perception field, influenced by sensor type and prevailing weather conditions. Additionally, the increased distance to obstacles may reduce perception accuracy, and blind spots created by structures like vehicles or buildings can further impede the comprehensiveness of their perception.

Cooperative perception empowers Connected and Automated Vehicles (CAVs) to achieve a comprehensive understanding of their surroundings, effectively eliminating blind spots in mixed traffic scenarios. Through collaborative activation with infrastructure, vehicles can gain the capacity to "detect" pedestrians. Consequently, vehicles can decelerate or stop promptly, especially if the pedestrian is crossing without paying attention to the road. The activation of cooperative perception stands as a valuable tool for reducing the occurrence of accidents.

This collaboration is particularly essential for the optimal performance of AV. By analogy, in a closed industrial context, each AIV has its embedded system that can allow it to communicate with other AIVs via the deployed network, or with the infrastructure where communication beacons can be fixed. They have also to watch out for collisions with pedestrians in warehouses, and navigate in changing environments. Future scenarios might involve AIVs going from one warehouse to another, as in an industrial zone, and therefore sharing roads with CAV.

In the research domain of CAV, the implementation of Cooperative-Intelligent Transport Systems (C-ITS) has become a central focus. Communication is the key of cooperative perception, and forms the backbone of collective intelligence, fostering its development and conception. In the Industry 4.0 context, cooperation between AIVs through communication can also contribute to having a greater autonomy of the fleet of AIV and therefore of each of its entities [Mí21].

V2X communication, illustrated in **Figure 1.3**, enables autonomous vehicles and infrastructure to exchange information about their surroundings.

C-ITS

As explained above, C-ITS plays a vital role in enhanced road safety (**Figure 1.3**). The **Figure 1.4** inspired by [Van+18] summarises where V2X communications can interfere in the various functions of an autonomous vehicle, whether for the road or industrial use.

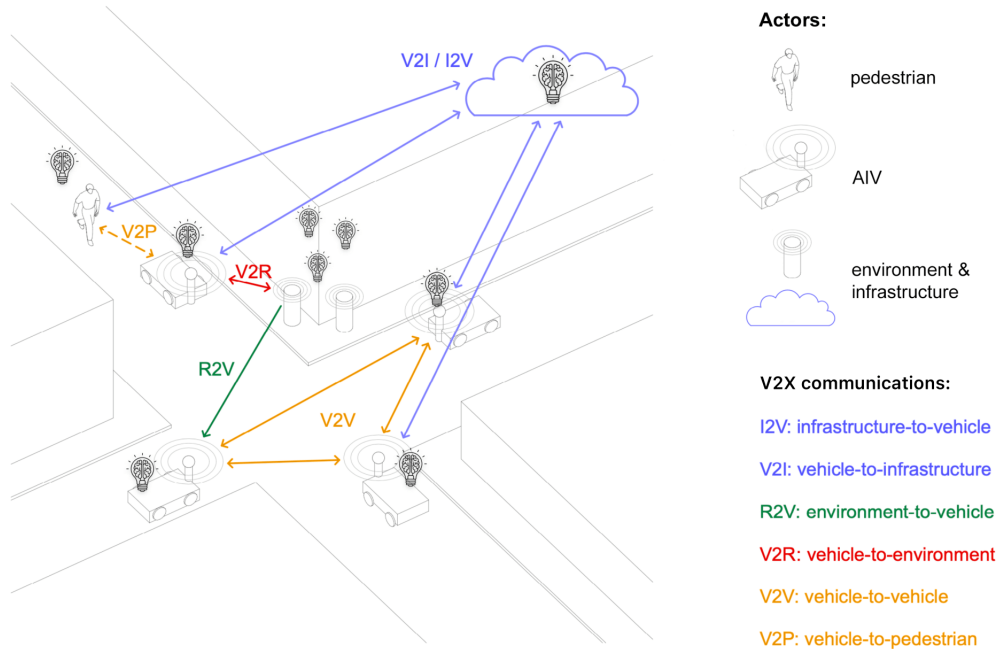


Figure 1.3 – Illustration of V2X communications

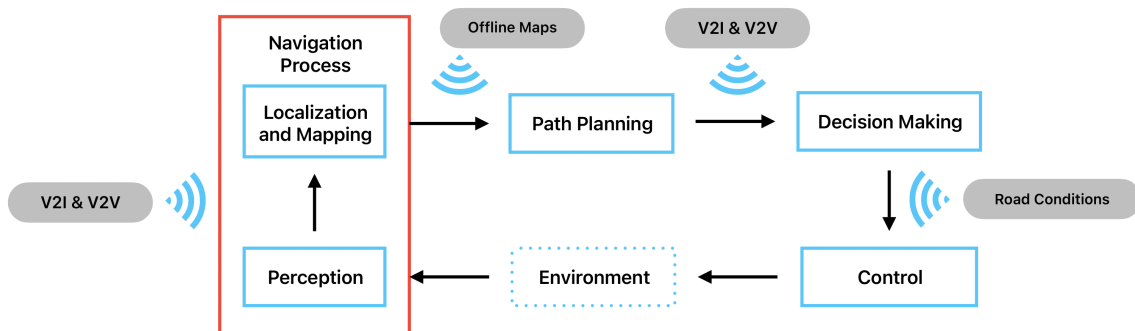


Figure 1.4 – Navigation process inspired by [Van+18]

Therefore, we are interested in the different types of messages standardized by European Telecommunications Standards Institute (ETSI) that allow communication between the different actors. [SCO16] and [C-R16] are European projects that aim to deploy C-ITS technologies in France. These V2X communications include different types of actors, shown in the **Figure 1.3**.

In Cooperative Intelligent Transport Systems (C-ITS), the different actors are:

- the ITS stations in the vehicles: On Board Unit (OBU)
- the fixed ITS stations deployed on the roads: Road Side Unit (RSU)

The ETSI provides standards for implementing the different messages and services of C-ITS. Inside CAVs and RSUs, there is a physical node where the RSUs system and services are installed. Some services need to be available as soon as deployed because they are defined as the Day 1 services in [C-R16]. Indeed, [CAR] explain that Day 1 services concentrate on information exchange to enhance predictive driving. Day 2 services elevate service quality through sharing perception and awareness data. From Day 3 onwards, services become more sophisticated, incorporating the sharing of intentions, support for negotiation, and cooperation, ultimately moving towards cooperative, accident-free automated driving.

Day 1 described four awareness messages including the Cooperative Awareness Message (CAM) (ETSI EN 302 637-2 standard [ITS14a]) and the Decentralized Environmental Notification Message (DENM) (ETSI EN 302 637-3 standard [ITS14b]). CAM contains information about the localization of the vehicle. Every C-ITS station will broadcast this message regularly, and then the location of all connected vehicles on the road will be known. On the contrary DENM are alert messages, issued at the time of an unexpected event in order to cooperate, notify and consequently disseminate information in the relevant geographical area.

The Road Works Warning (RWW) service broadcast information using DENM about the road works. This service is mainly focused on providing information and safety in road work areas because it warns the driver as it approaches.

In mixed traffic, CAVs need to be aware of vehicles even if they are not equipped with V2X facilities, as well as VRUs while they are making decisions. A first approach proposed in [Tsu+19] uses proxy CAM to enhance cooperative awareness in mixed traffic relying on RSUs. They are responsible for analysing the scene and for sending CAM on behalf of the non-cooperative entities (vehicle or VRUs).

To ensure greater cooperation between connected vehicles, the Day 2 applications

include solutions that use their sensors to scan the environment and share information. Day 2 applications [Lei+15] rely particularly on the Cooperative Perception Message (CPM) (ETSI standard TR 103 562 [ITS19]) to improve perception accuracy through cooperation between perceptive entities. It allows us to know the place of each non-connected object (e.g. VRUs), thanks to the sensors on CAV or infrastructure (e.g. cameras, LiDAR) [VTC24]. This exchange of information about the environment will enable CAV to see things that they cannot see directly.

Another way to cooperate is to inform other vehicles of these intentions. In this regard, the ITS Working Group 1 (WG1) is currently working on the definition of a Maneuver Coordination Service (MCS) and its Maneuver Coordination Message (MCM) [Cor+21]. The outcome of this work is in progress and is expected to be completed by the end of 2024.

Table 1.2 summarizes the V2X messages which will be used throughout this manuscript. Release 1 messages are designed to create awareness. However, Release 2 messages, such as CPM and MCM, can facilitate the initial stages of cooperation between vehicles. Although they do not include a cooperative interaction service, these messages offer an enhanced level of awareness by providing detailed perception information compared to Release 1 messages.

[ITS20] presents different infrastructure-specific messages which plays a crucial role in V2I (Vehicle-to-Infrastructure) communication. We resume three messages from this release in the **Table 1.3**.

Messages	Situations	Release
CAM	Send its position	Day 1
DENM	Breakdown, accident, traffic not fluent	Day 1
CPM	Perception of a static or dynamic obstacles (e.g. goods in the aisles, operators)	Day 2
MCM	Change of course (lane change), state directional intention to cross an intersection	Day 2

Table 1.2 – ITS messages used in the manuscript

Limitations

However, there is currently a limitation, many VRUs, such as pedestrians, lack C-ITS equipment. As a result, they cannot actively participate in this cooperative system, unlike

Messages	Descriptions
Infrastructure to Vehicle Information Message (IVIM)	Supports mandatory and advisory road signage such as contextual speeds and road works warnings
MAP Extended Message (MAPEM)	Represents the topology/geometry of a set of lanes
Signal Phase and Time Extended Message (SPATEM)	Disseminate the status of the traffic light controller, traffic lights and intersection traffic information
...	...

Table 1.3 – ETSI Infrastructure-specific messages [ITS20]

AVs. To address this and reduce vulnerable-vehicle accidents, various systems have been developed to assist drivers and road users.

For AVs, AI plays a crucial role in decision-making and leveraging cooperation for safety and efficiency. Other vehicles can be equipped with ADAS [Zie+16]. These systems rely on a comprehensive understanding of the vehicle’s surroundings, necessitating quick and precise processing of information for driver assistance. AI is increasingly integrated into these systems at various decision-making levels, requiring information for decision-making or generating elaborated information.

Regarding the status of the driver, it remains a critical factor. In non-autonomous or semi-autonomous vehicles, the driver must stay alert and ready to take control at any moment, especially in situations where the AI systems might face limitations or uncertainties. ADAS and other support systems are designed to enhance the driver’s capabilities and reaction time but do not eliminate the need for an attentive and engaged driver [EU20].

To assist drivers, CAVs also need a comprehensive understanding of their surroundings. To achieve this goal, the infrastructure, known as RSUs, gathers information from connected objects and scene descriptions provided by cameras. AI processes and fuses this information to enhance situational perception, sending it via V2X messages to other AVs. For the sake of simplicity, we will use the term AVs to refer to both CAVs and AVs throughout the manuscript.

The perception of local infrastructure, particularly from cameras, is crucial for ensuring safety in autonomous driving [Mao+23; Wan+23], facilitating the reintegration of VRUs into C-ITS. Improved perception contributes to better AV performance and enhances

ADAS algorithms. Before experimenting with various AI algorithms to fuse pieces of information collected in V2X messages and those provided by other sensors (e.g., cameras) in diverse scenarios, it is crucial to organize the collection of raw information. A lot of research on data fusion exists to aid the reintegration of VRUs. Indeed, this seems possible with the generation of video streams from infrastructure cameras, connected vehicles or other types of sensors (e.g. LiDAR).

However, it is important to specify that we want to process the video stream locally before sending the resulting information, rather than sending raw video streams. This local processing ensures that only relevant data is transmitted, which reduces bandwidth usage and enhances privacy and security by not transmitting potentially sensitive raw footage. Moreover, a remote entity may have difficulty processing raw video streams without knowledge about sensor quality, calibration, etc.

1.2.6 Energy management

Energy management is a critical aspect of AIVs because it directly impacts their operational efficiency and autonomy. As part of the five core functions (mentioned in **section 1.2.2**) AIV resource management—consists of monitoring battery status, technical errors/anomalies and conducting vehicle maintenance, as outlined by [KS19]. Efficient AIVs management necessitates a comprehensive approach, considering several factors such as operational availability, energy consumption [Las+14], collaboration between AIVs and the dynamic infrastructure, and their adaptation to changing conditions.

Energy management in AIVs involves achieving a balance between local and global needs to optimize efficiency and operational effectiveness. This balance can be understood through the concepts of local and global equilibrium in decision-making.

Local and global equilibrium

Local Equilibrium pertains to the individual vehicle’s perspective on energy management. Each AIV makes decisions based on its own battery status, current workload, and immediate recharging needs. Local equilibrium focuses on optimizing the vehicle’s own recharging schedule and energy consumption to ensure it remains operational within its specific context. For example, an AIV might decide to recharge if its battery level drops below a certain threshold, aiming to avoid running out of power during its next task.

In contrast, global equilibrium considers the entire fleet of AIVs and the environment

they operate within [DVS20]. It involves coordinating the energy needs of all vehicles to optimize overall system efficiency. Global equilibrium aims to balance the collective recharging demands across the fleet to prevent bottlenecks, reduce energy consumption, and improve overall operational efficiency. This can involve strategies such as scheduling recharges at off-peak times, sharing recharging infrastructure, and managing energy use based on the collective workload and availability of all vehicles.

To effectively manage energy resources, AIVs must find a balance between local and global equilibrium:

- Individual decision-making: Each AIV must be able to make informed local decisions about when and how much to recharge based on its current status and immediate needs [KS19]. This ensures that each vehicle operates efficiently and avoids running out of energy.
- Collective coordination: Simultaneously, AIVs need to coordinate with each other and with the infrastructure to achieve a global equilibrium. This involves sharing information about their energy status and upcoming recharging needs to avoid situations where too many vehicles require recharging at the same time.

Indeed, a primary objective in energy management is to optimize recharging cycles based on energy costs, to prevent inflated energy consumption due to frequent recharges during low workload periods. In addition, poor anticipation can significantly limit system availability. The irregular distribution of AIV missions, which includes periods of intense activity and quieter intervals, necessitates correlating energy consumption with workload and operational availability.

Achieving optimal balance between local and global decision

Achieving an optimal balance between local and global decision-making in energy management for AIVs is essential for maintaining efficiency and sustainability. To prevent an overload of simultaneous recharging requests, AIVs need to collaborate through inter-communication or via the infrastructure. Although automatic recharging resolves the issue of the number of charges, it requires space and consumes energy. Even marginal reductions in energy consumption (2-3% of reduction), are significant for industries such as warehouses and airports. Thus, for the introduction of AIVs' fleet in the industry of the future, it therefore seems necessary to fine-tune the number of recharging points. This optimization can be refined by considering communication capabilities among AIVs, enabling collective avoidance of critical (urgent) recharging instances.

To address these challenges and achieve an optimal balance, the three key points that stand out are:

- Communication and collaboration: By enabling communication between AIVs, vehicles can share their energy status and recharging needs. This allows for collective decision-making and coordination, reducing the likelihood of simultaneous recharging requests and optimizing the use of available recharging points.
- Infrastructure management: The recharging infrastructure itself must be managed to support both local and global needs. This includes planning for sufficient capacity and strategically locating recharging stations to meet the collective demands of the fleet while accommodating individual vehicle needs.
- Adaptive strategies: The energy management system must be adaptive, allowing it to respond dynamically to changes in workload, vehicle availability, and energy consumption patterns. This adaptive approach helps maintain a balance between local vehicle requirements and global fleet efficiency.

In summary, effective energy management in AIVs involves balancing local and global considerations. Local equilibrium focuses on optimizing the efficiency of individual vehicles, while global equilibrium addresses the collective needs of the fleet to enhance resource utilization and infrastructure capacity. By integrating communication, collaboration, and adaptive strategies, AIVs can achieve a more efficient and sustainable energy management system.

Optimization techniques

Reducing energy consumption has become a critical issue across all sectors, making optimization an indispensable tool. Therefore, the choice of the cost function is crucial. Various mathematical models are defined for mobile robots to address different objectives of the problem. The power function is frequently used because it depends on simple parameters such as speed and motor force [MM30; Zha+21].

Both reducing and increasing speed have a significant impact on energy consumption, and variations in motor force also affect energy usage. The speed profile generator is determined by the motor force, which depends on the acceleration (or deceleration) and the mass of the mobile robot. Optimizing energy consumption involves finding an optimal control strategy modelled by the motor force.

Several techniques have been employed to address the optimization problem in mobile robots. One prominent approach is the use of genetic algorithms. For instance [Šte+21]

propose a genetic algorithm to reduce the energy consumption of a mobile robot. The cost function is the sum of the linear velocity squared and angular velocity squared. This algorithm relies on an optimal fuzzy logic controller, which enables the robot to track different paths (e.g., zigzag, sharp turns).

Another technique involves the fuzzy logic Mamdani, as demonstrated by [GT22]. This method determines three outputs: curve speed, energy consumption, and time information. Using optional curved path data (radius, center angle, load, ground friction), three fuzzy rules are defined to identify these outputs. The total energy consumed is the sum of the power sensor, control, and motion. A trapezoidal speed profile is utilized to create an optimal profile based on fixed times or distances for straight paths, involving three segments: acceleration, constant speed, and deceleration. Alternatively, a triangular speed profile can be used, requiring only acceleration and deceleration phases. Minimum energy consumption is achieved from these speed profiles for straight paths. For curved paths, the speed profile is derived from fuzzy logic, incorporating the maximum and minimum values of the trapezoidal speed profile. The integration of these methods highlights the importance of optimizing speed and motor force to minimize energy consumption for mobile robots.

In the railway sector, a technique based on braking distance is commonly used. [Tan+18] address the problem of train speed trajectory optimization as an optimal control problem. The cost function represents the net electrical energy, which depends on the maximum electrical traction and braking forces, as well as the efficiency of electric motors during traction and braking operations. The authors propose a numerical algorithm based on the Pontryagin's Maximum Principle (PMP) [Kop62] to find the optimal speed trajectory. Their study identifies conditions for traction and braking phases, showing the effectiveness of this approach through comparisons with other numerical optimization methods.

For mobile robots, the goal is to solve an energy optimization problem specific to their environment (e.g. warehouse, airport). The cost function is defined by the integral of the power delivered by the robot's motor, reflecting the absolute work done. This refinement of the cost function offers a more accurate reflection of the energy consumption during the robot's mission. This approach enables the robot to perform acceleration, deceleration, and constant speed movements simultaneously, offering a more accurate representation of energy consumption during its mission.

Improving the cost function allows for more precise energy management strategies,

tailored to the unique demands of each environment. When applied to a fleet of AIVs, this approach offers several key benefits:

1. **Optimized energy usage:** Each AIV can manage its energy consumption more effectively based on its specific operating conditions and movement patterns.
2. **Better coordination:** A refined energy model allows AIVs to work together more effectively, reducing energy waste. For instance, AIVs can schedule their operations and reloads strategically. If one AIV is running low on battery, the system can coordinate with other AIVs to ensure it recharges at the optimal time, such as during periods of low activity or when other AIVs are handling tasks, thereby avoiding unnecessary downtime and ensuring continuous operation, thereby improving the autonomy of AIVs within the fleet.
3. **Fleet-level efficiency:** Collective energy management helps in balancing the load across the fleet, reducing the likelihood of energy shortages or excessive battery depletion in any single vehicle.

Overall, a comprehensive approach to energy management not only improves the performance and reliability of each AIVs but also enhances the overall efficiency and operational effectiveness of the entire fleet in various industrial settings.

1.3 Collective mobile robot autonomy

Collective mobile robot autonomy involves the coordination and cooperation of multiple robots to achieve a common goal. This section explores three key aspects: mobile robot cooperation, collective intelligence, and collective autonomy.

1.3.1 Mobile robot cooperation

Mobile robot cooperation is fundamental to the development of autonomous systems that can work together efficiently in an environment. This cooperation enables multiple robots to perform tasks collaboratively, sharing information and coordinating their actions to achieve common objectives. Several key technologies and methodologies have been developed to facilitate this cooperation in the road field.

For many years, the scientific community has focused on platooning and Cooperative Adaptive Cruise Control (CACC) [Xin+21]. These technologies enable vehicles to

maintain inter-vehicle distances and reduce energy consumption by responding to the maneuvers of leading vehicles (e.g., slowing down, braking) with minimal delay. Effective communication with very short delays significantly enhances the efficiency of cooperative autonomous vehicles, especially for convoy traffic in industrial environments. While platoon organization is well understood and efficient [Liu+23], it demands stability of convoys over time, which can be a limitation.

This is why it is interesting to look at less constrained modes of cooperation such as CACC. Several V2X communication technologies and CACC mechanisms have been proposed in the road domain. The constraints of an industrial environment are different, it is necessary to evaluate the operation of these mechanisms using tried-and-tested proven methodologies [Lya19] but which need to be adapted.

Similarly, other cooperative manoeuvres have been studied in the literature. Track insertion [Seq+19] can be adapted to the industrial case that interests us in the project by employing messages similar to those defined for the road environment. Finally, beyond cooperation on manoeuvres [Ron+18], cooperative perception makes it possible to exchange already elaborated information about pedestrians and to identify potential areas of conflict with road vehicles [NY22], as discussed in **subsection 1.2.5**.

[Zha+23] propose four different classes:

- **Class A:** Cooperative Perception - "Tell each other where I am and what I see" (status sharing)
- **Class B:** Cooperative Prediction - "Tell each other this is and what I plan to do" (intent sharing)
- **Class C:** Cooperative decision-making - "Let us do this together" (agreement seeking)
- **Class D:** Infrastructure-enabled decision-making - "I will do as directed" (prescriptive)

1.3.2 Collective intelligence

"A **collective intelligence** is a large Multi-Agent System (MAS) where:

1. There is little to no centralized communication or control.
2. There is a provided world utility function that rates the possible histories of the full system." [WT99]

The concept of collective intelligence extends the idea of cooperation among robots from achieving individual objectives to creating an intelligent whole. In this framework, robots work together not just to fulfill individual tasks but to contribute to the intelligence and efficiency of the entire fleet. The fleet, as a cohesive entity, becomes intelligent through the collective actions and interactions of its individual members.

Our description of collective intelligence in robotic fleets is that each robot's actions are informed by the collective's state and objectives, allowing for dynamic and adaptive responses to changing environments and tasks. This approach not only enhances the efficiency and effectiveness of individual robots but also optimizes the performance of the fleet as a whole. For instance, robots can share raw sensor data and predictions, leading to improved situational awareness and decision-making capabilities for the entire group.

In the ITS field, special topics of automated vehicle coordination and/or cooperation, using Vehicle-To-Vehicle (V2V) or Vehicle-To-Infrastructure (V2I) communication techniques, are well studied. [RM17] cover the literature related to the coordination of connected and automated vehicles for intersections or merging at highway on-ramps, using centralized and decentralized approaches, with the goal of limiting traffic congestion and reducing transportation energy consumption and emissions by improving traffic flow.

[MCZ21] propose a taxonomy and a survey on coordination of AVs to treat traffic problems like intersection management, smart parking, ride-sharing, ramp merging or platooning. In the particular case of crossing intersections, [Glo+21] propose an adaptive approach capable of selecting the most appropriate solution presented by [MCZ21], depending on the traffic situation.

In essence, AIVs systems benefit from the principles of collective intelligence by enhancing coordination and adaptability through shared information. This integration not only improves individual performance but also optimizes overall system efficiency, addressing complex challenges like traffic management and autonomous operation.

1.3.3 Collective autonomy

In the field of ITS, particularly within the context of smart cities [CAF17], the autonomy of vehicles has been comprehensively categorized by the SAE into six levels of autonomous driving [Wis22]. These levels, described in **subsection 1.2.1**, are widely recognized for assessing the capabilities of AVs. However, this structured framework is not developed for AIVs, with research in this area remaining relatively sparse [EBW16].

These levels provide a clear framework for assessing the progression of vehicle auton-

omy. However, they focus primarily on the individual vehicle’s ability to perform tasks like navigating an intersection without collision. Yet, they do not address the efficiency of these maneuvers—for instance, whether the vehicle can complete the task smoothly or if it requires multiple stops. This limitation is particularly critical when evaluating the efficiency of public transport systems within the road sector.

The challenge becomes even more pronounced in industrial settings, where the scale of operations is significantly larger. When managing not just two pallet trucks but fleets of fifty or more AIVs, the ability to assess and optimize fleet efficiency becomes essential. The question then arises: how can we effectively evaluate the performance of an entire fleet of AIVs?

To address the limitations of the existing autonomy scale, particularly for AIVs, our approach emphasizes the need for decentralized solutions and collective strategies within fleets. Enhancing the autonomy of AIVs requires moving beyond individual decision-making to foster cooperation among all vehicles in the fleet [CFK96; HS19; RAT19]. By increasing both individual and collective intelligence, AIVs can improve their performance in core tasks: navigation process (perception and localization), TA, path planning, motion planning and vehicle management.

Given the potential of collective strategies, we will propose the development of a new scale of autonomy that reflects the integration and cooperation within an AIV fleet. AIVs can execute their missions more effectively through a collective strategy, where the fleet communicates and shares information. To begin with, we propose classifying V2X messages presented in **Table 1.2** according to the four classes of cooperation [Zha+23] mentioned in **subsection 1.3.1** (see **Table 1.4**).

	Class A	Class B	Class C	Class D
Messages	CAM DENM CPM	MCM	ACK_MCM	SPATEM MAPEM IVIM

Table 1.4 – V2X messages classified according to the four classes of cooperation

At the foundational level, the proposed scale encompasses basic cooperative behaviours such as collective obstacle avoidance and the sharing of localization and environmental perception data. As we progress to more advanced levels, the scale incorporates real-time TA, scheduling, and distribution among the fleet, thereby improving overall efficiency, minimizing downtime, and optimizing energy usage. At its highest level, the scale ad-

dresses the collective management of energy resources across the fleet, ensuring that all vehicles operate in a coordinated and energy-efficient manner.

Building upon this classification, we propose initiating a new hierarchical scale of autonomy that extends beyond individual vehicles to consider collective performance. By shifting focus from isolated tasks to the integrated operation of a fleet, we aim to develop a framework that better reflects the complexities and demands of large-scale AIV deployments. This scale delineates progressive levels of collective point of view for a fleet of AIVs, emphasizing the cooperative capabilities of the fleet. The levels range from no collective autonomy to advanced optimization, integrating various V2X messages to facilitate each stage. The messages written in cyan in the autonomy scale below will be proposed to solve problems of maneuver cooperation at intersections ([ACK_MCM](#)) and collective TA ([CTM](#), [CRM](#)) in the following **chapters 2** and **3**, respectively.

Levels of autonomy from a collective point of view for a fleet of AIVs:

- **Level 0:** No collective autonomy
 - Vehicles operate independently without sharing information or coordination.
- **Level 1:** Homogeneous robots - collective mapping and collision avoidance
 - Vehicles share and create a dynamic map of their environment.
 - Vehicles share location data to avoid collisions.
- **Level 2:** Heterogeneous robots - interoperability and cooperative awareness and perception
 - Vehicles share and create a dynamic map of their environment.
 - Vehicles share location data to avoid collisions.
 - Vehicles and infrastructure devices with different capabilities and communication protocols coordinate effectively.
 - Ensures seamless communication and collaboration across various systems.
 - V2X messages used: CAM, DENM, CPM
- **Level 3:** Collective decision-making and collective TA
 - Vehicles collaboratively decide on maneuvers.
 - V2X messages used: CAM, DENM, CPM, MCM, [ACK_MCM](#), [CTM](#), [CRM](#) (Class A, B, and C messages for perception, intent, and decision-making).
- **Level 4:** Optimisation
 - The fleet carries out the skills of the other levels with an emphasis on economy and collective reduction in energy impact.
 - Vehicles collaborate and optimize actions for overall efficiency, including energy

consumption.

- V2X messages used: All of the above (CAM, DENM, CPM, MCM, [ACK_MCM](#), IVIM, SPATEM, MAPEM), utilizing full Class A, B, C, and D messages for comprehensive perception, prediction, decision-making, and following infrastructure directives.

A more advanced and accurate hierarchical autonomy scale can be particularly useful for industrial applications where the efficiency and coordination of large fleets of vehicles are essential. In industrial environments, where operations often involve the management of dozens or even hundreds of autonomous vehicles, this scale provides a structured framework for evaluating and improving collective performance. By grading autonomy levels from basic cooperation to advanced fleet-wide optimisation, industries can systematically improve the way they interact, allocate tasks and manage resources such as energy. This not only improves the fluidity and efficiency of operations but also reduces downtime and operational costs, ensuring that the fleet is operating at peak performance. In addition, scale helps to identify specific areas where technological advances or process optimisations are needed, directing investment and development efforts towards those areas that will deliver the best results in terms of productivity and operational efficiency.

Future research on collective autonomy for AIV fleets will prioritize collaborative decision-making processes among AIVs operating in shared environments. In this manuscript, we explore how to progressively integrate various collective functions into an AIVs fleet. Our focus is on developing collective strategies for obstacle avoidance, TA, perception, and energy management, with the goal of improving the overall efficiency and adaptability of AIV fleets.

To conclude, the contributions proposed in this manuscript will be classified according to the new hierarchical scale of autonomy for AIVs that we have outlined. By using this scale, we will demonstrate how our contributions align with the various levels of collective autonomy, from basic cooperative behaviours to advanced fleet optimization. This approach ensures that the solutions we develop are systematically integrated into the broader context of AIV deployment, ultimately enhancing the efficiency and effectiveness of autonomous vehicle fleets in complex operational environments.

1.4 Methodologies and tools used to simulate and experiment of AIVs

The advancement of AIVs marks a significant milestone in the evolution of intelligent systems, requiring sophisticated tools and methodologies to ensure their effective development and deployment. As AIV systems become increasingly complex, the demand for robust simulation and experimentation techniques becomes critical. These techniques are essential for the validation of AIV systems but also play a central role in the iterative processes of design, testing, and optimization.

This section explores the various methodologies and tools employed in the simulation and experimentation of AIV systems, offering a comprehensive overview of current best practices and emerging trends in the field. It highlights the strengths and limitations of different approaches, emphasizing that simulation tools must be carefully adapted to the specific problems being addressed, as well as to the solutions used to tackle these challenges. Moreover, the techniques applied should be tailored to ensure the system's performance is accurately assessed and optimized.

1.4.1 Simulation in the field of AIVs

"**Simulation** is a key technology for developing planning and exploratory models to optimize decision-making as well as the design and operations of complex and smart production systems. It could also aid companies to evaluate the risks, costs, implementation barriers, impact on operational performance, and roadmap toward Industry 4.0" [FAS20]

In the field of AIVs, simulation allows us to take into account the different constraints and requirements formulated by manufacturers and prospective users of these AIVs. Before starting on large-scale testing of AIV traffic scenarios in especially complex industrial or airport situations, simulation stands as a primary step before deploying [HZ05]. A significant advantage lies in the ability to generate usable results without necessitating the application of scaling factors.

The main benefits of simulating AIV operations are extensively presented by [TBS18]:

- reduces the development time and cost of an AIV,
- minimises the potential operational risks associated with the AIV,
- enables the feasibility of different AIVs scenarios to be assessed at a strategic or operational level,

- provides a rapid understanding of AIV operations (under conditions of limited data availability),
- identifies improvements in facility layout configurations hosting AIVs; the simulation also provides flexibility in terms of deployment and redeployment, and
- enables us to study the sharing of responsibility between the central server and the robots (local/global balance) for the various operational decisions.

A further advantage of simulations is to introduce humans into scenarios. This helps instill confidence among stakeholders regarding the safe coexistence and potential interactions between forthcoming mobile robots and human operators in industrial settings, before a future deployment [Hen+19].

1.4.2 Agent-based systems

"An **agent** is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators" [RND10]

This definition underscores that AIV can be modelled as an agent. The agent paradigm is widely used to simulate or model complex, interactive, adaptive, distributed or cooperative systems and has become commonplace [Mac16]. Agent-based systems have been proposed in many engineering domains, including AV [Jin+20]. The distribution of agents (decentralization) enables these systems to be flexible. Additionally, the agent concept is particularly well-suited for modelling and simulating cyber-physical systems [VLL15].

Therefore, agent-based simulation is a significant tool for advancing algorithmic development before deployment. Thus, we will review the literature on agent concepts, agent-based models, principles of agent simulations and MAS.

Agent concepts

The concept of an agent is based on three pillars, shown in **Figure 1.5** deployed in an environment equipped with sensors and knowledge that can interact through communication and coordination in various organizational structures, including centralized, decentralized, or hierarchical organisation systems.

Developing an agent architecture for the simulation of AIV traffic situations is important. Each agent within this architecture possesses its knowledge and has functional capabilities for observation, communication, decision-making, and action.

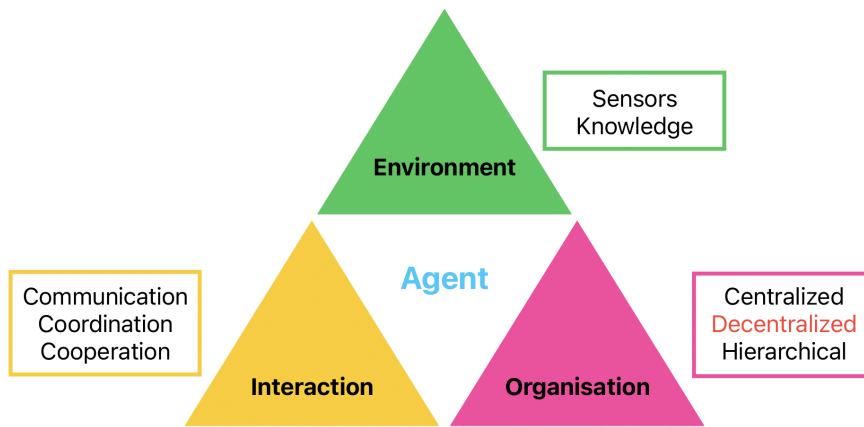


Figure 1.5 – Agent properties

In the context of Industry 4.0, the requirements of flexibility, robustness and scalability lead to the consideration of decentralized mechanisms to react to contingencies. Before being definitively adopted, decentralization must first be modelled and simulated.

[RAT19] reviewed the literature and presented two different types of classification for agents, summarised in **Table 1.5** and **Table 1.6** with definitions.

Type of agents	Behaviour
Reactive	React to environmental changes Sense and act
Deliberative	Initiate actions without any external trigger and rely on planning Sense, Plan and Act
Hybrid	Maximize a utility function

Table 1.5 – Type of agents [RAT19]

"Hybrid agents perform actions based on a planning algorithm or react to current perceptions." [RAT19]

Throughout this manuscript, AIVs will be defined as hybrid agents in this manuscript. Indeed, because of their dual capacity for planning and reaction. These vehicles operate in dynamic and often unpredictable industrial environments where strategic planning and real-time reactivity are essential. AIV use planning algorithms to map out optimal paths and sequences, ensuring efficiency and adherence to overall operational objectives.

At the same time, they rely on sensor data to perceive their immediate environment and execute reactive behaviours to manage unexpected obstacles, changes in the environment, or emerging safety issues. Additionally, because AIVs are integrated into a fleet, they can interact with other AIVs through communication and coordination. This integration allows, for example, cooperative perception, enabling AIVs to share information about their surroundings and optimize their actions collectively.

This hybrid nature enables AIVs to balance long-term operational goals with the agility required for real-time adaptation, enhancing their flexibility, robustness, and scalability in the context of Industry 4.0 environments.

Moreover, psychologists have worked on modelling the task and the operator, including [Ras83] who proposed a 3-level model. Researchers [Fou02] have drawn inspiration from this to create an agent model depicted in **Figure 1.6**. An agent can have a behaviour adapted to the task it is carrying out and therefore does not necessarily have a fixed behaviour.

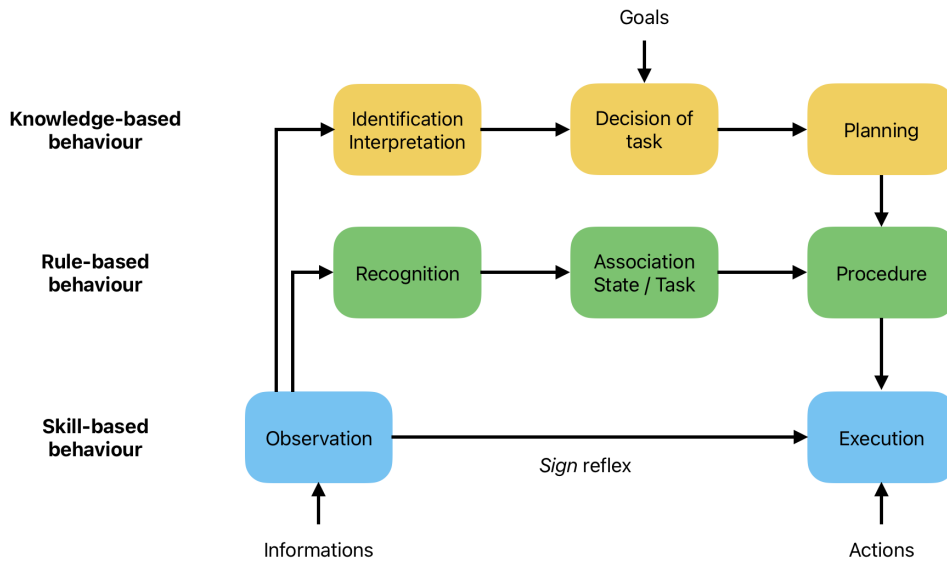


Figure 1.6 – Rasmussen’s model of the operator: three levels of behaviour adapted from [Ras83]

Then, [RND10] introduced a refined categorization for "learning" agents, distinguishing between those that learn elements of the environment and those that adjust their control algorithms’ parameters with the guidance of a critic. **Table 1.6** resumed it.

AIV integrated into a fleet that performs collective strategies to improve the autonomy of the fleet must exhibit characteristics of multiple types of learning agents. Firstly, they

Type of learning agents	Behaviour
Simple reflex	React to current sensory input
Model-based reflex	Keep an internal state of the environment
Goal-based	Perform actions to complete a goal
Utility-based	Maximize a utility function

Table 1.6 – Type of learning agents [RND10]

have to demonstrate aspects of simple reflex agents by reacting to current sensory input, such as avoiding collisions or adjusting their speed based on immediate environmental cues.

Secondly, AIV can embody model-based reflex agents by maintaining an internal representation or state of the environment, enabling them to anticipate and respond to recurring patterns or events. Additionally, AIV have to operate as goal-based agents, performing actions aimed at achieving specific objectives or completing tasks efficiently within the industrial setting.

Finally, AIV may incorporate aspects of utility-based agents by making decisions that aim to maximize a utility function, which could involve optimizing routes to aid minimize its energy consumption and therefore of the fleet too.

Agent-based modelling simulations

Agent-based modelling offers a cost-effective and resource-efficient approach to modelling complex systems due to the flexibility, autonomy, and scalability it provides to agents [DKJ18]. The structure of an agent-based model has been defined by [MN10], as follows:

1. **Agents:** the individual entities with their characteristics, and behaviours.
2. **Interactions:** the relationships among agents and the ways they interact, defined by a network of connections determining how and with whom they interact.
3. **Environment:** the surrounding context within which agents operate and interact, influencing and being influenced by the agents.

The article [Mac16] presents definitions on Agent-Based Modelling Simulation (ABMS) across various categories, as depicted in (**Figure 1.7**). A common characteristic among all these definitions is agents are defined as individual heterogeneous entities.

The resulted definitions in **Table 1.7** can be seen as a scale of agent autonomy, ranging from individual agents with prescribed behaviours to adaptive agents capable of changing

<i>ABMS definition</i> / <i>Agent properties</i>	Behaviours	Interactions	Adaptability
Individual	Prescribed, scripted	Limited	None
Autonomous	Autonomous, dynamic	Limited	None
Interactive	Autonomous, dynamic	Between other agents and the environment	None
Adaptive	Autonomous dynamic	Between other agents and the environment	Agents change behaviours during the simulation

Table 1.7 – Definitions for ABMS based on agent properties [Mac16]

behaviours during simulation. Our goal to improve the autonomy of AIV integrated into a fleet, executing collective strategies, aligns with the modelling of adaptive AIV agents.

Before delving into scenario testing, the goal is to transition towards a modelling framework that encompasses adaptive ABMS. This approach facilitates the exploration and optimization of collective strategies that empower AIV to dynamically adjust their behaviours in response to the evolving environment and operational requirements.

By conceptualizing AIV as adaptive agents, the aim is to approach experiments with the behaviour of AIV adapting in real-time, optimizing their actions in response to coordination, communication and therefore evolving circumstances. Finally, moving closer to the simulation of adaptive agents is in line with the objective of evaluating collective strategies that maximise the efficiency, robustness and adaptability of AIV fleets operating in Industry 4.0 environments.

Multi-agent simulation

As described above, a MAS can be a network of autonomous entities called agents, each capable of independent decision-making and learning. These agents collaborate by interacting with each other or their environment to solve tasks [DKJ18].

MAS can be centralized, decentralized, or hybrid, and may involve physical robots called multi-robot systems in the literature. The study [RAT19] gives a state-of-the-art of the categorisation of cooperative heterogeneous multi-robot systems. **Table 1.8** resumed the different classifications based on agent diversity and communication capabilities, interaction complexity, and types such as cooperative, competitive, or collaborative.

Several ABMS tools have been elaborated to model agents-based systems utilizing frameworks such as UML or Agent-UML [Bau+01]. Then, the MAS modelled, the objective is to simulate complex scenarios with interactions among agents in an environment.

Agents' diversity	Agents' communication capabilities	Agents' interaction complexity	Type of interaction between agents
Sensory	Homogeneous non-communicative	No direct interaction	Cooperative interaction
Actuation capabilities	Homogeneous communicative	Some interaction	Competitive interaction
Cognition algorithms	Heterogeneous non-communicative	Complex condition interaction	Collaborative interaction
	Heterogeneous communicative		

Table 1.8 – The different categories based on multiple criteria of MAS [RAT19]

A study conducted by [DKJ18], outlines three of the most widely used simulators in MAS to evaluate methods for agent-based systems. These simulators are:

- **Java Agent Development framework (JADE)**: open-source platform based on Foundation of Intelligent Physical Agents (FIPA) standard for building MAS in Java language. It provides a comprehensive set of tools and libraries for creating, deploying, and managing autonomous agents [BCG07].
- **GAMA**²: modelling and simulation platform with the support of GAML (high-level agent-based language).
- **Matlab**: software that is primarily known for its general-purpose computational capabilities, it is also utilized to analyze the performance of MAS. It can be integrated with JADE for in-depth investigations into MAS performance, such as for smart distribution networks [Elm+15].

1.4.3 Advanced decision-making techniques for AIVs

To effectively enhance the autonomy of mobile robots, it is important to explore a range of advanced decision-making techniques that improve navigation and obstacle avoidance capabilities. [Pan17] provide a comprehensive review of mobile robot navigation and obstacle avoidance techniques, highlighting several advanced methods including neural networks, neuro-fuzzy systems, genetic algorithms, Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO) algorithms [Pan17]. Each of these techniques has specific strengths that make them suitable for different aspects of autonomous navigation and decision-making.

Neural networks excel in learning and adapting to complex and dynamic environments, which makes them effective for navigation in unpredictable settings. For instance, [Che+22] proposed a neural network model to address autonomous navigation challenges

2. GAMA Platform: <https://gama-platform.org/>

in environments with obstacles.

Neuro-fuzzy systems combine the learning capabilities of neural networks with the handling of imprecision by fuzzy logic, neuro-fuzzy systems enhance decision-making in uncertain environments. This approach is explored by [Mis+22], who developed a system that integrates deliberative and reactive navigation control, providing corrective decisional commands to the robot.

Genetic algorithms use natural selection principles to find optimal paths in complex environments, as shown by [LBE18], who applied them to path planning in static environments.

PSO inspired by the social behaviour of animals, explores search spaces efficiently, leading to faster and often optimal solutions. For example, [ZWC19] developed an integrated localization approach combining a local search-based PSO algorithm with the particle filter for robust and accurate global localization.

Additionally, ACO based on the foraging behaviour of ants, is effective for solving pathfinding and routing problems, as illustrated by [Mia+21], who applied ACO to indoor mobile robot path planning.

Some research focuses on the use of fuzzy agents to handle the levels of imprecision and uncertainty involved in modelling the behaviour of simulated vehicles [Fou13]. Indeed, fuzzy set theory is well suited to the processing of uncertain or imprecise information that must lead to decision-making by autonomous agents [FO13].

Building on this foundation, a promising approach to improving the performance of AIVs involves the application of fuzzy logic. By leveraging fuzzy logic, AIVs can better handle varying levels of uncertainty and make more accurate decisions, thereby enhancing their overall efficiency and effectiveness.

1.4.4 Fuzzy logic systems

Today, most of the control tasks presented in the **subsection 1.2.2** (*localization, mapping in the navigation process; TA and motion planning; decision-making and motion control*) performed by autonomous mobile robots have been the subject of performance improvement studies using fuzzy logic:

- navigation of mobile robots from conceptual, theoretical or application points of view [HNK12], navigation of several mobile robots [PPP09], navigation and control of a mobile robot in an unknown environment in real-time [Yer+15], and comparison of navigation performance of mobile robots obtained using fuzzy logic or neural

- networks [Yud+19];
- obstacle avoidance from conceptual and systemic points of view in an unknown dynamic environment [Mey+18];
- path planning strategies focusing on obstacle avoidance [Shi+18] or global navigation [Pat+19];
- motion planning [NC18] ;
- localization of mobile robots [AOH20];
- intelligent management of energy consumption [LN22].

The use of agents having fuzzy knowledge can manage the levels of imprecision and uncertainty [Ros09], involved in modelling the behaviour of simulated vehicles.

Fuzzy agent concepts

An agent-based system can be fuzzy if its agents have fuzzy behaviours or if the knowledge they use is fuzzy. This means that agents can have:

1. fuzzy **knowledge**: fuzzy decision rules, fuzzy linguistic variables, and fuzzy linguistic values;
2. fuzzy **behaviours**: the behaviours adopted by the agents as a result of fuzzy inferences;
3. fuzzy **interactions, organisations or roles** [OFF12].

Fuzzy agents can follow the evolution of fuzzy information coming from their environment and from the agents [ÖG31]. By interpreting the fuzzy information they receive or perceive, fuzzy agents interact within a MAS; they can also interact in a fuzzy manner.

For example, a fuzzy agent can discriminate a fuzzy interaction value to evaluate its degree of affinity (or interest) with another fuzzy agent [OFF12]. The different elements of a fuzzy agent model are as follows [Fou13]:

- the agent-based fuzzy system;
- the behaviour of a fuzzy agent, inspired by perceive-decide-act feedback loops [Bru+09];
- the behavioural functions of a fuzzy agent;
- the fuzzy interactions between two fuzzy agents.

Fuzzy logic simulation

To design controllers, a lot of research utilized "numerical computing environments" such as:

- [Mey+18] used Matlab: to compare T1FLS (Type 1 Fuzzy Logic System) and ITS2FLS (Interval Type 2 Fuzzy Logic System) for obstacle avoidance of mobile robots in different environments. A type-2 fuzzy logic controller using type-2 fuzzy sets can handle such uncertainties to produce a better performance [Hag04].
- [Shi+18] used Matlab: to design fuzzy logic controllers that emerge as suitable for mobile robot navigation because of robustness and ability to handle uncertainties in static unknown environment.
- [NC18] used Matlab: to propose two separated fuzzy logic controllers to develop an intelligent algorithm in order to avoid both static and dynamic obstacles.
- [Yud+19] used Scilab: to investigate the comparison of fuzzy logic controllers and neural network applications in mobile robot navigation.

These studies demonstrate the effectiveness of fuzzy logic controllers and other intelligent algorithms in handling uncertainties and improving mobile robot navigation in various environments. Moreover, for example, Matlab offers a comprehensive set of fuzzy logic functions through its Fuzzy Logic Toolbox, allowing users to design, simulate, and analyze fuzzy inference systems with ease.

On the other hand, FisPro (Fuzzy Inference System Professional) [GC11] exists which is a free and open-source specialized software dedicated to fuzzy inference systems. This one offers a more focused approach allowing users to rapidly construct and evaluate fuzzy inference systems without the need for extensive programming knowledge.

1.4.5 Emulators and simulators to move towards real experiments in the field of AIVs

Before conducting real experiments in autonomous mobile robotics, we are exploring physics rendering simulators in two crucial domains. This preliminary step allows us to thoroughly test and refine our strategies, ensuring greater efficiency and safety when we transition to real-world scenarios.

Specifically, we focus in this section on communication, traffic, and physics rendering simulators to enhance the autonomy of AIV using collective intelligence strategies. To facilitate testing various scenarios before deploying them in real-world experiments, we

seek simulators capable of incorporating V2X communications. Our review spans both the robotics perspective and the road domain, which has made significant advancements in autonomous vehicles.

Communication and traffic simulators

In the field of robotics, Robot Operating System 2 (ROS2)³ has emerged as a significant middleware for communication and simulation. It is an open-source framework for the development of robotics applications and is a tool favoured by researchers and even industrialists today. ROS2 facilitates communication between various robotic components using a publisher-subscriber model, which is essential for simulating complex robotic interactions.

It supports real-time control systems and large-scale distributed architectures [MKA16]. Compared to ROS, there is no master entity, and ROS2 utilized Data Distribution Service (DDS). It makes it highly suitable to work with for developing and testing autonomous distributed robotic systems [Eró+19], for example, cooperative algorithms for AIV with V2X communications. Moreover, it can be integrated with simulation environments like Gazebo⁴ which will be presented below in the section.

In the road field, [AM13] summarises the different popular road traffic and network communication simulators. It classified them into 3 types:

- **network communications** simulators
- **road traffic** simulators
- **combined** simulators: communications and traffic

Network communication simulators are able to reproduce the behaviour of communication technologies such as OMNeT++⁵, [vargaandrasDiscreteEventSimulation2001] and NS-3⁶, which simulate V2X communications but not in a specific environment. On the contrary, Simulation of Urban Mobility (SUMO) [Lu+20] is for simulating multi-modal road traffic from real-world maps, but these do not allow V2X communications to be analysed. Other simulators, combine the simulation of traffic and network communication, which seems interesting but their combination is not suitable for our research. This is the case of Vehicles in Network Simulation (VEINS) [SGD11] combining SUMO and OMNeT++.

3. ROS: <https://www.ros.org/>

4. Gazebo: <https://gazebo.org/home>

5. OMNeT++: <https://omnetpp.org/>

6. NS3: <https://www.nsnam.org/>

These simulators enable the sending and receiving of C-ITS messages and provide a detailed analysis of transmission performances, including delays and losses, depending on network load and radio propagation model. By integrating with a traffic simulator like SUMO, it becomes feasible to consider C-ITS messages and implement C-ITS mechanisms to impact vehicle behaviour. However, this approach lacks the ability to interact with the environment beyond C-ITS, especially in terms of perception (e.g., camera, LiDAR). Additionally, it does not allow for studying interactions with a human driver, unlike a dedicated driving simulator.

Robotics simulators

With a view to a futuristic Industry 4.0, where robots could share roads with connected or unconnected vehicles, the value of a simulator adapted to robotics is being considered.

Moreover, in our research problem, we want to analyse video streams to detect VRU and communicate information from CAV and local infrastructure. In this case, AV can react to interactions and message exchanges. In particular, this would enable humans to drive in the simulator, receive notifications and interact with ADAS. It could be used to assess the acceptability of such a system. They will be able to take the messages into account, and this contribution to the simulator could be used to test the ADAS, the related algorithms and their acceptability by the driver.

One of the best-known open-source simulators is Gazebo. It can be used to simulate robot systems in complex environments. However, it does not simulate the behaviour of AVs, nor does it provide a realistic rendering for training AVs.

Realistic simulators to train autonomous road vehicles

Realistic simulators are produced with game engines such as Unity or Unreal Engine. They make it possible to test specific problems with realistic rendering scenarios in the globality and obtain the associated V2X communications. This is not possible with representative autonomous driving datasets existing, created for a specific problem (e.g. vision for KITTI [Gei+13]). Even, nuScenes [Cae+20] which is the first dataset to carry the full AV sensor suite built with Carla Simulator [Dos+17]. This dataset makes use of a vehicle with a full 360-degree field of view using 6 cameras, 5 RADARs and 1 LiDAR.

Indeed, one of the research challenges we are addressing is enhancing the cooperative perception of AIVs. To achieve this, it is essential to find solutions for testing V2X data fusion in simple scenarios such as intersections. This could enable connected vehicles, with

the help of the local infrastructure, to improve their overall perception and reintegrate pedestrians not perceived by their own sensors.

Therefore, one possibility with realistic simulators which seems possible is the opportunity to obtain V2X communications generated in a realistic scenario through the analysis of data flows (video, LiDAR) issued from CAVs or infrastructure sensors (camera, LiDAR, RADAR).

AWSIM⁷ is a Unity-based digital twin simulator tailored for autonomous driving research, created for the use of Autoware [Kat+15; Kat+18]. This one is an open-source software for self-driving vehicles that can be combined with AutoC2X [Tsu+20]. This system enables cooperative perception by using OpenC2X [Lau+16] for Autoware-based AV. This solution is focused on the operation of an AV. Studying the fusion of V2X data in a complex scenario with the scaling of communications to traffic levels seems complicated. On top of that, it is more a co-simulator than a traffic AV simulator.

On the contrary, the simulator Carla [Dos+17] can be used with a Cooperative Driving Autonomation (CDA) known as OpenCDA [Xu+21a; Xu+23]. This framework is composed of three modules:

- a co-simulation platform with simulators,
- a full-stack cooperative driving system, and
- a scenario manager

Carla Simulator associated with OpenCDA can provide the possibility to obtain different information from sensors and implement algorithms to modify behaviours of AVs induced by interaction messages. Moreover, it will allow us to get different information for the content of V2X messages, as well as the possibility of using the "Scenario Manager", with realistic driving scenes (weather conditions, geographical environments). We use the integration of Carla and OpenCDA, as described in [VTC24], to generate V2X messages using a specific architecture with ROS2. This setup creates a simulated V2X communication network between AV and RSU in the Carla environment. This contribution will be detailed in **chapter 4**.

7. AWSIM: TIER IV inc. - <https://tier4.github.io/AWSIM/>

1.5 Conclusion

In this chapter, we delved into the integration of AMRs within the paradigm of Industry 4.0, highlighting the numerous challenges and opportunities for improving the autonomy of a fleet of AIV with collective strategies.

On the one hand, in **section 1.2**, we review the current landscape of AIV integration in the Industry 4.0. Their integration requires a shift towards decentralized intelligence, where each vehicle can operate independently while contributing to collective goals. By enhancing the autonomy and decision-making capabilities of individual AIV, and fostering collective intelligence across fleets, industrial environments can achieve greater efficiency, adaptability, and safety, particularly in dynamic and complex settings.

Main functions performed by AIVs

After this overview, we outline the main functions performed by AIVs: navigation process (localization and perception), TA, motion planning, decision-making and control (see **Figure 1.2**), as foundational to autonomous navigation. However, the increasing complexity of industrial environments and the need for seamless integration into dynamic systems highlight the limitations of current models. Enhancing these functions with more sophisticated algorithms and adaptive systems is essential for improving the autonomy and efficiency of AIVs, particularly in large-scale operations. Afterwards, we reported the different advancements in the literature on these core functions.

In summary, **obstacle avoidance** is a critical component of autonomous navigation for AIVs, encompassing a range of strategies from perception and detection to motion planning and collision avoidance. Effective obstacle avoidance requires a holistic approach that integrates mapping, localization, and communication between vehicles to handle both static and dynamic obstacles. Obstacle avoidance strategies have evolved to incorporate sophisticated techniques for detecting and navigating around obstacles. However, the increasing complexity of industrial operations, where AIVs must navigate not only static but also dynamic and unpredictable obstacles, exposes the limitations of existing methods. By leveraging advanced localization techniques, sophisticated sensors, and cooperative algorithms, AIVs can enhance their autonomy, ensuring safe and efficient navigation in complex environments. To address these challenges, we will extend Bhanes' algorithm [BKH16] in **chapter 2 - section 2.2**, as detailed in [ARCI22; ICAE23], to better manage the complexities of dynamic obstacle environments. The exploration of these elements

lays a foundation for further advancements in autonomous systems, with the potential to significantly improve the operational capabilities of AIVs in industrial settings. The exploration of these elements lays a foundation for further advancements in autonomous systems, with the potential to significantly improve the operational capabilities of AIVs in industrial settings.

Cooperative perception, inspired by advancements in the automotive sector, offers a promising solution by enhancing situational awareness through V2X communication. Yet, integrating this into industrial AIVs requires overcoming significant challenges, particularly in terms of sensor fusion, data sharing, and interaction with non-equipped entities. The advancements in cooperative perception within the road sector, particularly for CAVs, hold significant promise for application in industrial environments. The challenges faced by AIVs in terms of perception, navigation, and safety share many parallels with those encountered by road vehicles, such as sensor limitations and the need to avoid collisions with VRUs. Cooperative perception enhances the ability of vehicles to understand and respond to their environment by leveraging V2X communication, enabling them to share critical data and reduce blind spots. This technology, crucial for road safety, has the potential to significantly improve the autonomy and safety of AIVs in industrial settings as well, particularly in scenarios where they interact with human workers or navigate complex environments. The use of C-ITS standards, including awareness and perception messages like CAM, DENM, and CPM, provides a framework for these interactions, facilitating a higher level of situational awareness and cooperation among vehicles and infrastructure. While challenges remain, particularly in integrating non-equipped VRUs into this system and ensuring effective data fusion, the local processing of sensor data before transmission presents a viable solution to these issues, preserving bandwidth and privacy. Overall, the integration of cooperative perception into industrial automation not only enhances the performance and safety of AIVs but also aligns with broader Industry 4.0 goals, paving the way for more intelligent, connected, and autonomous industrial operations.

TA plays a critical role in optimizing the efficiency of mobile multi-robot systems, such as AIV fleets. The taxonomy of TA provides a structured approach to assigning tasks based on factors like the type of robots, the nature of the tasks, the modality of allocation, and the degree of interdependence between tasks. The objectives of TA, whether focused on cost, behaviour, reward, priority, or utility, guide the development of solutions, ranging from optimization algorithms to market-based mechanisms. Furthermore, the complexity

of mobile robot task planning underscores the importance of both the decomposition of tasks and the strategic assignment to maximize system performance. However, despite these advances, existing centralized approaches often lack the flexibility, robustness and scalability required in dynamic environments such as Industry 4.0 or Airport 4.0. These limitations highlight the need for decentralized methods, in particular those based on market-based concepts, as promising routes to achieving efficient TA in autonomous and adaptive systems. In addition, current messages that would enable standardized communication are designed for autonomous vehicles that therefore have individual goals and not a common goal. They are therefore not adapted to the collective TA processes required in heterogeneous fleets of AIVs. To overcome these shortcomings, in **chapter 3** we present a dynamic strategy in cooperation mode with the infrastructure, proposing two new messages for coordination and communication in TA processes within various fleets.

Another core function is **energy management** which is a pivotal component in the operational success of AIVs, directly influencing both their efficiency and autonomy. While current techniques, such as optimization algorithms and adaptive recharging strategies, provide a solid foundation, they are often insufficient in addressing the full scope of challenges presented by large-scale and high-demand environments. The balance between local and global decision-making in energy management is particularly difficult to achieve, often leading to inefficiencies or bottlenecks in the system. This calls for further innovation in energy management strategies, including more advanced optimization techniques and better integration with fleet-wide operations.

Moreover, considerable research attention is increasingly directed towards TA and vehicle management, reflecting their growing significance in deploying larger fleets. Our point of view to improve the autonomy of AIVs is to move towards decentralized solutions with collective strategies taken by the fleet of vehicles. To increase their intelligence and their individual and collective decision-making, AIVs need to become more cooperative [CFK96; HS19; RAT19]. Understanding and improving these functions is crucial for enhancing the collective autonomy and efficiency of AIVs in industrial applications. Thus, the AIVs will be more efficient in performing the core tasks.

Collective mobile robot autonomy

In **section 1.3**, we explore the literature about collective mobile robot autonomy which represents a sophisticated evolution in autonomous systems, emphasizing not only the capabilities of individual robots but also their ability to function cohesively as a fleet.

This section discussed the critical components of this paradigm, including mobile robot cooperation, collective intelligence, and collective autonomy.

Mobile robot cooperation is essential for achieving seamless coordination among multiple robots. By utilizing technologies such as platooning and cooperative adaptive cruise control (CACC), robots can work together efficiently, sharing information and synchronizing their actions to achieve shared goals. The application of various communication mechanisms, such as V2X technologies, enables robots to perform complex maneuvers and adapt to diverse operational environments.

Building on this foundation, collective intelligence extends the concept of cooperation by enhancing the overall intelligence of the robot fleet through collaborative interactions. By sharing data, predictions, and decisions, the fleet collectively improves situational awareness and decision-making capabilities. This approach not only boosts the performance of individual robots but also optimizes the operation of the entire group, leading to better traffic management and energy efficiency in various contexts, from road networks to industrial settings.

The combination of mobile robot cooperation and collective intelligence highlights the significant potential of collective autonomy. In this manuscript, we aim to enhance collective autonomy for AIVs, enabling better integration and cooperation of AIVs within a fleet. The autonomy scale defined by SAE for AVs, addresses individual vehicle capabilities. However, for industrial applications involving large fleets, a new scale of autonomy that accounts for collective performance and cooperation is necessary. This proposed hierarchical scale ranges from basic cooperative behaviours to advanced optimization of fleet-wide actions, including TA, obstacle avoidance, and energy management with the use of V2X messages to be adapted for heterogeneous fleet of AIVs. By evaluating and improving collective autonomy, industries can achieve greater efficiency, reduced operational costs, and enhanced overall performance.

Methodological framework

In the manuscript, we will set up a methodological framework consisting of modelization, simulation and experimentation of cooperative algorithms solutions to problems of autonomy for industrial vehicles. Simulation allows the constraints and requirements formulated by manufacturers and future users of AIVs to be taken into account. Simulation provides a good framework for studying solutions to these different challenges. Therefore, in **section 1.4**, we provide a comprehensive overview of simulation and experimentation

tools employed in robotics, emphasizing the significance of robust simulation techniques such as agent-based, neural networks, genetic algorithms and fuzzy logic simulations. These methodologies provide a foundational understanding for developing and testing advanced robotic algorithms in controlled, yet realistic, environments.

To effectively simulate our proposed algorithms and collective strategies, we will utilize the methodology of ABMS. This framework will enable us to model the complex interactions and behaviours within the MAS, providing a robust platform for evaluating the effectiveness of various strategies. We will develop various multi-agent simulations in Python, to test our solutions for various scenarios. It will give us complete control over the development process. This is particularly important for managing communication between the various agents representing the AIVs in our simulations. This approach offers flexible adaptation to the various aspects of AIVs management and facilitates any adjustments required for deployment on the industrial site.

Additionally, fuzzy logic will be employed in specific contributions, particularly in energy management scenarios (in **chapter 5**), where the inherent uncertainty and complexity of decision-making processes can be better addressed through this approach. By combining ABMS with fuzzy logic, we aim to create a comprehensive simulation environment that captures both the dynamic and uncertain nature of the systems we are studying.

Furthermore, we discuss a wide range of simulators that exist for both robotics and AVs research employed to transition from simulation to real-world experimentation. We focused on simulators encompassing aspects such as network communication, traffic, and hybrid approaches. Additionally, we introduce robotics and realistic simulators developed to train autonomous road vehicles. In summary, many simulators have limitations when it comes to integrating comprehensive communication capabilities with realistic environmental interactions. Specifically, the simulators discussed vary in their ability to simulate complex traffic scenarios, realistic sensor data, and cooperative communication strategies such as V2X. Although simulators like Gazebo are well-suited for robotics applications, they lack the realism required for training autonomous vehicles. Conversely, simulators such as SUMO and VEINS are excellent for traffic and communication simulations but do not offer the necessary integration with advanced perception systems.

Given these limitations, we have chosen to utilize Carla Simulator, OpenCDA, and ROS2 for our research. This combination offers the most comprehensive platform for simulating realistic traffic environments, enabling V2X communications, and facilitating

the development and testing of cooperative autonomy strategies for AIVs. In particular, Carla, in conjunction with OpenCDA, will allow us to generate V2X messages and simulate complex driving scenarios, while ROS2 will provide the communication infrastructure necessary for real-time interactions. This setup will be further elaborated in **chapter 4**.

To conclude, this thesis hence aims to propose cooperative algorithms, and strategies to enhance various functions of AIV thereby improving the collective intelligence of a fleet of AIVs. The various contributions of this thesis regarding collective strategies, specifically in obstacle avoidance, TA, cooperative perception, and energy management, are detailed in the following chapters.

COLLECTIVE OBSTACLE AVOIDANCE STRATEGIES

2.1 Introduction

The deployment and integration of AIVs in industrial settings face several challenges, including employee acceptance, vehicle localization, traffic fluidity, and the perception of disturbances within dynamic environments. The current autonomy of AIVs is often constrained by predefined trajectories and limited awareness of their surroundings. To enhance the efficiency and safety of AIV operations, it is essential to explore strategies that improve their autonomy through better information exchange and collective intelligence.

AIV fleets operating in complex industrial environments, such as storage areas and production lines, must adapt to evolving traffic constraints, dynamic and often unpredictable conditions. Traditional approaches, which rely on static paths and limited situational awareness, may not suffice in these scenarios. Obstacle avoidance, a critical aspect of AIV motion control, becomes increasingly complex when multiple AIVs operate simultaneously in the same space. Effective communication among AIVs, as well as with infrastructure, is vital for seamless coordination and obstacle avoidance. Given the complexities of operating multiple AIVs in shared spaces, real-time information exchange and collective decision-making are crucial.

In this chapter, we explore innovative strategies that leverage collective intelligence to enhance obstacle detection and avoidance capabilities in AIVs. These strategies aim to:

1. **Enhance adaptation to traffic constraints:** Collective strategies enable AIVs to dynamically adapt to changes in their environment, ensuring smoother operations even as conditions evolve. This adaptability is crucial for managing the fluidity of traffic in environments like warehouses and production lines, where conditions can shift rapidly.

2. **Improve decision-making:** When faced with incomplete or fragmented information, AIVs that operate collectively can make more informed decisions. By sharing data and insights, AIVs can compensate for individual knowledge gaps and uncertainties, leading to more effective obstacle avoidance and overall safety.
3. **Facilitate communication:** Effective communication between heterogeneous AIVs and with surrounding infrastructure is essential for coordinated actions.

The chapter is organized as follows: **section 2.2** presents a contribution of a collective strategy to a collision detection algorithm. Then, **section 2.3** proposed a resolution of global obstacle avoidance by AIV with a collective strategy. **appendix C**, presents a method for estimating the positions of AIV moving in a closed industrial environment. Finally, **section 2.4** discusses the emulation of an intersection scenario in an industrial setting to test the proposed detection algorithm, in order to do future real-world experiments to validate the effectiveness of the strategies.

This chapter is related to several publications: [[ARAC24](#); [ARCI22](#); [ASPAI22](#); [ICAE23](#)].

2.2 Contribution of a collective strategy to an obstacle avoidance algorithm

As previously mentioned, the objective is to enhance obstacle avoidance for AIVs within a fleet. One approach is to leverage existing algorithms as a foundation allowing for iterative improvement and innovation. With this idea, we will propose to improve an algorithm for obstacle avoidance in order to integrate a cooperative solution. The methodology employed in this section involves developing or augmenting an algorithm, proposing an agent-based model, and rigorously testing these solutions through simulations before progressing to emulated and real-world experiments in **section 2.4** of this chapter. By following this structured approach, we aim to develop and validate enhanced obstacle avoidance strategies for AIVs, ultimately contributing to safer and more efficient industrial operations.

In this section, we will begin by recalling the limitations of BAHNES' algorithm [[BKH16](#)] and then our proposition of extension. Then, we will propose an explanation of the ABMS used, before presenting the simulation environment and scenarios. Finally, we will analyse simulation results before discussing them.

This section is linked to the following publications: [ARCI22; ICAE23].

2.2.1 Algorithm improvement

The collision avoidance algorithm of [BKH16] presented in the **section 1.2.3** makes it possible to deal with the priority of different vehicles when approaching an intersection. We will call their algorithm: the Bahnes' algorithm throughout the manuscript.

This strategy proposes a cooperative approach for determining the priority of AIVs at intersections through message exchange. The method requires each vehicle to know its location and be capable of communicating with other vehicles. This collision avoidance algorithm enables AIV to interact and cooperate using various message types.

The communication among AIVs involves three distinct message types:

- **Hello_msg**: used by an AIV to announce its presence position to other AIVs;
- **Coop_msg**: sent by an AIV before reaching an intersection to coordinate and determine priority with other AIVs.
- **Ack_msg**: used by AIVs to acknowledge the receipt of a **Coop_msg** from another AIV.

However, as previously mentioned, this algorithm does not tackle the issues of detecting, communicating with, and avoiding both stationary and moving obstacles (such as human operators). Therefore, our contribution focuses on ensuring that missions are completed successfully without being disrupted by obstacles along the planned route, we propose an extension to Bahnes' algorithm. The dotted red area of the **Figure 2.1** illustrates the proposed extension to Bahnes' algorithm, which is depicted in black.

Our extension enhances the original communication protocol among AIVs by introducing mechanisms to detect, communicate, and avoid both stationary and dynamic obstacles commonly encountered in warehouse environments. Specifically, we introduce two new message types for collaborative perception:

1. the **Obstacle_msg**: sent by an AIV to other AIVs circulating in the warehouse to indicate the perception of an obstacle,
2. the **Alert_msg**: sent by an AIV to alert other AIVs in the warehouse about the presence of an unavoidable obstacle.

When an AIV detects an obstacle, it broadcasts an *Obstacle_msg* to nearby AIVs. The AIV then determines whether the obstacle is dynamic. If the obstacle is dynamic, the algorithm evaluates the urgency of the AIV's task. If the task is urgent, the AIV

emits an auditory signal to alert both the obstacle and surrounding AIVs. If the task is not urgent, the AIV waits for the obstacle to clear.

For stationary obstacles, or when avoidance is necessary, the AIV decelerates and navigates around the obstacle. If the obstacle cannot be avoided, the AIV sends an *Alert_msg* to notify other AIVs of the risk of collisions or delays, ensuring coordinated navigation. This extension preserves the integrity of the original message types — *Hello_msg*, *Coop_msg*, and *Ack_msg* — while significantly enhancing the algorithm’s robustness in dynamic, real-world operational settings.

To further enhance the extension of BAHNES’ algorithm, the *Obstacle_msg* can be augmented to include additional critical information. Along with indicating that an obstacle is unavoidable, the message could also convey details such as the estimated increase in delay (i.e., the extra cost) and the impossibility of safely crossing paths with other AIVs in the alley. This enhanced communication could enable nearby AIVs to better assess the impact on their routes, facilitating improved decision-making regarding rerouting, task prioritization, and overall traffic management. Ultimately, this could contribute to a more efficient and coordinated navigation system within the warehouse, minimizing the risk of delays or collisions.

2.2.2 Agent-based model simulation

To explore the extension of BAHNES’ algorithm within the framework of the three scenarios proposed by [BKH16], we establish an agent-based model.

As previously mentioned in **chapter 1, section 1.4.2**, agent-based simulation for AIVs [TBS18] is the most common approach in the same way as simulations based on discrete events or robotics software [Mac16]. AIVs possess message exchange capabilities and LiDAR systems enabling them to detect vehicles ahead.

For instance, given an AIV agent \tilde{v}_i , if another AIV agent \tilde{v}_j in front of it is stopped or travelling at a slower speed, the AIV agent \tilde{v}_i can detect it with its LiDAR and stop accordingly to avoid hitting it, as shown in **Figure 2.3**.

We assume that individual autonomy facilitates fleet deployment and operation, however sharing some information would increase the responsiveness of each agent. Thus, increasing the collective autonomy of the AIV agents would strengthen the decision-making, and the individual autonomy of each AIV agent.

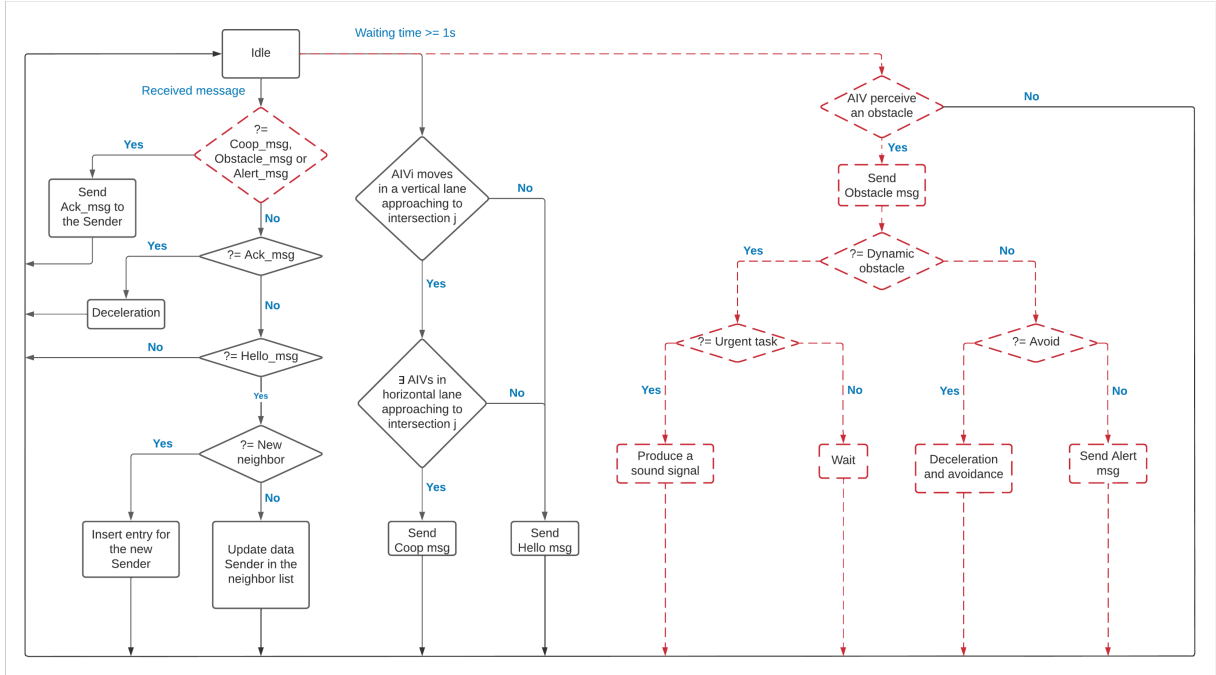


Figure 2.1 – Improvement of Bahnes’s algorithm to deal with the correlated problem of collision and obstacle

Agent-based model

Figure 2.2 presented the agent-based model proposed to test our collective obstacle avoidance strategies for AIVs in simulation. The model uses color-coding to distinguish between different classes: red represents dynamic elements, green indicates static elements, and purple is used for elements not linked to the environment.

The AIV navigates along a predefined pathway referred to as the *Circuit*. This *Circuit* outlines the routes or pathways that AIVs follow and can vary in form depending on the specific application context. It is a critical component of the AIV’s operational environment, guiding its movement and ensuring it adheres to a set path. This adherence is vital for tasks such as material handling, transportation, or automated operations in industrial and controlled settings.

The objective is to have an ABMS designed generically to test different scenarios, but also different types of circulation plans. Thus, the *Circuit* model is extendable to more concrete models, such as 4 – *LaceCircuit* or *IndustrialCircuit*. In this chapter, we will use these two types of circuits, respectively, presented in section 2.2.3 and section 2.3.2.

Indeed, **Figure 2.2** illustrates that an infrastructure is deployed in the environment and is composed of a circuit, and active elements such as beacons, tags and stations. These three active elements are modelled as agents. AIV, which are also represented as agents operating according to predefined missions defined by path on the traffic map. However, static or dynamic obstacles (e.g. operators) may be present in the environment.

Therefore AIV agents are equipped with LiDAR to detect pedestrians, other AIVs (dynamic obstacles) and goods (static obstacles) present in their activity area. Then, to move in their environment and accomplish their mission, they have knowledge about their environment through their own perception of the environment and through the information received from other AIVs.

Moreover, they can also cooperate by exchanging information with other AIVs or even with the infrastructure thanks to different types of standardized messages: CAM, DENM, CPM, MCM and Acknowledgment Maneuver Coordination Message (ACK_MCM). This allows AIVs to build up their own dynamic map of the environment. Thus they are cooperative, pro-active and autonomous to carry out their missions without colliding with static and dynamic obstacles.

Finally, **Figure 2.2** does not detail AIV localization. This aspect of the model is further developed and discussed in **appendix C**, where a computational approach is presented for determining the AIV's next position on the 4 – *LaceCircuit* based on its current location.

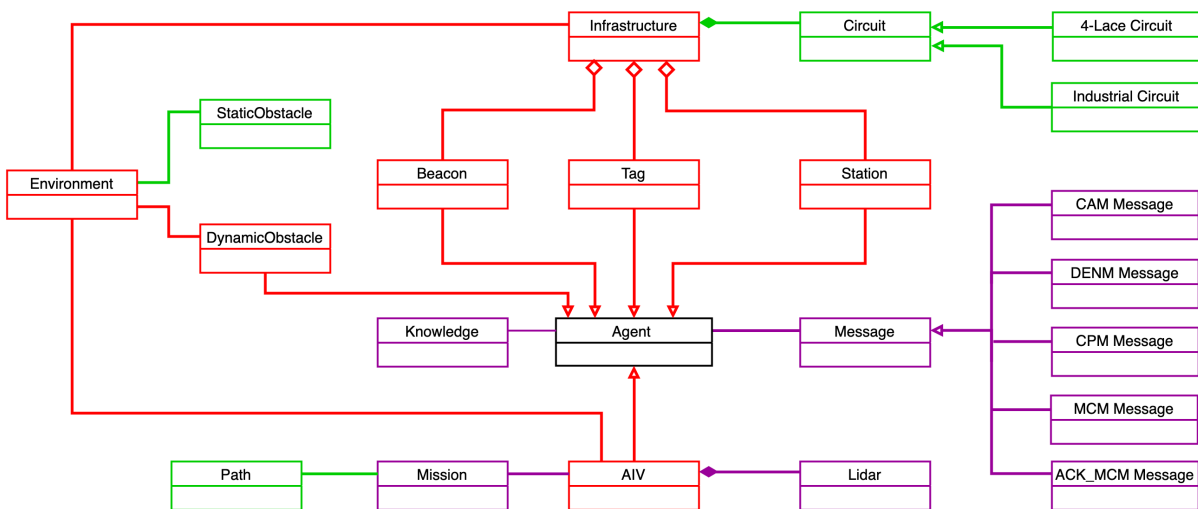


Figure 2.2 – Agent-based model for AIVs obstacle avoidance

2.2.3 Environment simulation and scenarios

When designing the initial scenarios involving AIVs, our focus was particularly on situations that favour crossing traffic. Accounting for this characteristic led us naturally to diagram the circulation in four loops referred to as the 4 – *LaceCircuit* (shown in **Figure 2.3**). This configuration is specifically designed to test the augmented Bahnes algorithm. Moreover, it allows us to launch scenarios that we consider as a benchmark plan for comparing results in this **section 2.2**.

We will simulate two different scenarios. In both scenarios, four AIV agents continuously circulate in the environment, moving independently (while exchanging messages to cross intersections) and their speeds change randomly (see **Figure 2.4.a**). The simulations are carried out using the same specifications and constraints. The only difference is in the second scenario (Sc_2), where obstacles are randomly generated on the circuit (spatial and temporal generation).

To enhance visualization in illustrations, each AIV agent has its own color (orange, blue, green and red). When an AIV approaches an intersection, it first detects the intersection using a specific tag system (denoted as Tag_i). Each intersection has its own unique tag, allowing the AIV to identify exactly which intersection it is approaching.

For instance, in **Figure 2.4.b**, when the green and red AIVs approach the intersection, they send a *Coop_msg* to the other AIVs. After receiving a request, the other AIVs sends an *ACK_msg* to show its agreement, as shown in **Figure 2.5**. A priority list of intersections known to the AIVs is then updated after all AIVs have agreed. Thus, an AIV that receives unanimous agreement first is placed at the top of the priority list and can therefore afford to cross. In **Figure 2.5**, the green AIV sends its request first, securing agreements before the red AIV. Consequently, in **Figure 2.4.b**, the green AIV crosses the intersection while the red AIV waits for its turn.

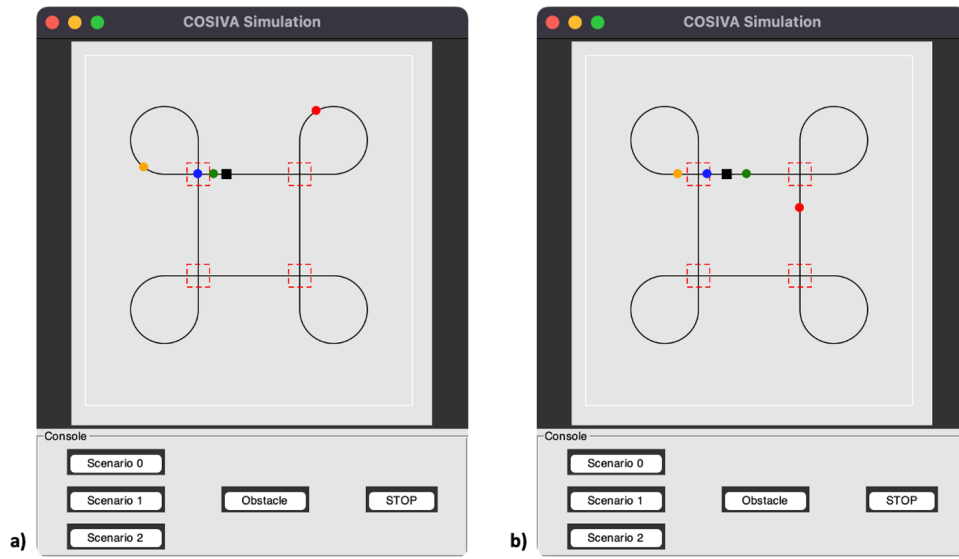


Figure 2.3 – Simulation of LiDAR use: a) at the top of the picture: a green AIV perceives a fixed obstacle in front of him; while waiting for the green AIV to avoid the obstacle, the LiDAR of the blue AIV allows him to stop and keep its distance to avoid colliding, b) the green AIV avoided the obstacle, and the blue AIV perceives, in turn, the obstacle

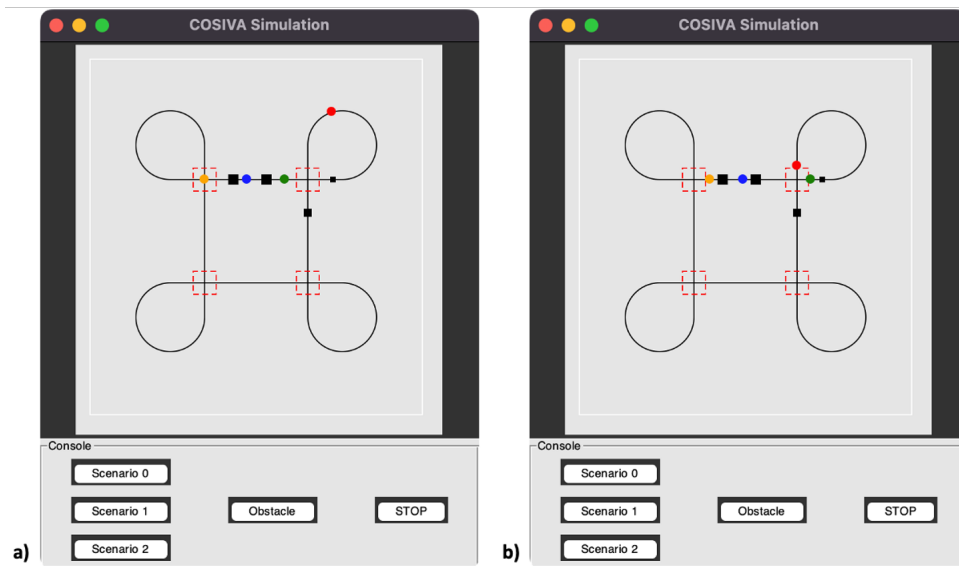


Figure 2.4 – Simulation of the Bahnes' algorithm: a) on the right top side of picture two green and red AIVs arrive at an intersection, b) the green AIV passed the intersection after communicating with other AIVs, and the red AIV waits to cross the intersection

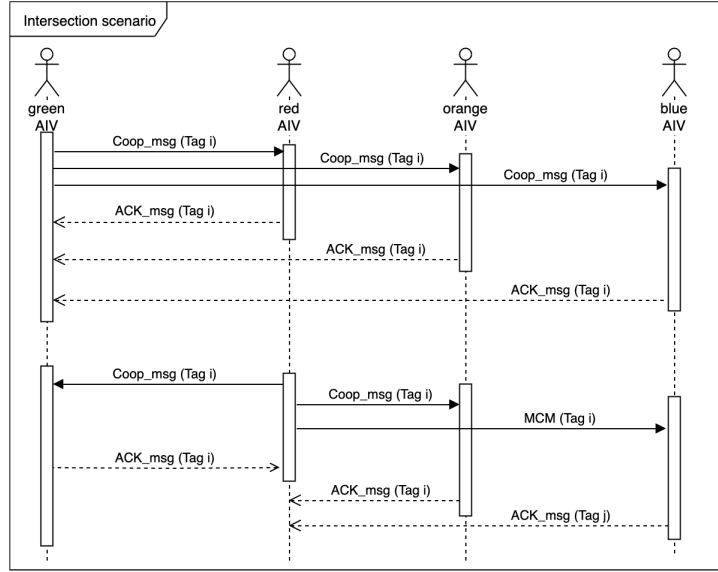


Figure 2.5 – Bahnes’ algorithm used for intersection crossing corresponding to **Figure 2.4**

2.2.4 Simulation results

The simulations of **Scenario 1** (Sc_1) on the circuit 4 – *LaceCircuit* with four AIVs circulating at different speeds made it possible to verify the absence of collisions when using the extended Bahnes algorithm. We were able to measure the cost of implementing this algorithm on the AIVs, with the objective of maximizing the number of complete laps performed by each AIV.

Table 2.1 summarises the results:

- 31% less of complete laps for the AIV that circulates the fastest, which is the result of the many slowdowns at very frequent intersections – nevertheless, this is still superior to an AIV at average speed and in nominal conditions (without other AIV on the circuit, or obstacles);
- 15% less of complete laps for an average speed, which becomes acceptable for the implementation of anti-collision between AIVs – the cost in a number of complete laps performed is less and the energy expenditure is much lower since the speed variations (speed reductions then accelerations to cross the intersections) decrease significantly compared to the AIV moving at a higher speed;
- 13% less of complete laps for a lower speed, which represents a small gain compared

to the average speed and which means that it is not necessary that the AIVs circulate too slowly.

AIV	Velocity	Number of turns / 15mn nominal conditions	Number of turns / 15mn anti-collision algorithm	Anti-collision algorithm cost
AIV_1	5	39	27	31%
AIV_2	4	13	11	15%
AIV_3	4	13	11	15%
AIV_4	3	8	7	13%

Table 2.1 – S_{C_1} : anti-collision algorithm simulation results

In S_{C_2} , obstacles are generated randomly on the circuit (spatial and temporal generation). The LiDAR of an AIV agent can also perceive obstacles in the aisles ahead that constrain its path. On perceiving them, it cooperates to warn other AIVs by sending an *Obstacle_msg*, and then avoids the fixed obstacle by going around it, if possible, as in the situation (see **Figure 2.3** and **Figure 2.4**).

As discussed in **chapter 1, subsection 1.2.3**, we will classify the obstacles detected by AIVs based on their size (s_1 to s_4) and impact on navigation as follows:

- s_1 or s_2 : These are obstacles that, while requiring the AIV to slow down, can still be avoided without necessitating a change in its path. Their size may vary.
- s_3 or s_4 : These are obstacles that the AIV cannot bypass without rerouting. Encountering such obstacles forces the AIV to turn around and find an alternative path, with sizes that may differ.

Table 2.2 provides the results of these simulations, in a number of complete laps performed by the four AIVs, according to varying numbers and sizes of obstacles. Given the circuit chosen for these tests, a number of obstacles greater than ten do not seem to make sense. On the other hand, the sizes of the obstacles are classified from 1 to 4 in ascending order of their encumbrance on the circuit.

Table 2.2 shows that for the same number of obstacles, the impact on the number of turns made by the AIVs will depend on the size of these obstacles. Thus, considering three obstacles, the AIV_1 will do 23 laps if the obstacles are small or medium (sizes: s_1 , s_2 , and s_3), while it will only do 19 laps if they are large (sizes: s_4 , s_3 and s_4). The incidence is much lower on the two AIVs having an average speed, and not remarkable for the AIV_4 moving at low speed.

If we vary the number of obstacles with sizes of similar values (here four, five then ten obstacles), the incidence only becomes slightly significant for ten obstacles. The

probability of having ten obstacles (even a group of humans) on this small circuit at the same time is very low, we can consider that this is a very good result.

Table 2.3 provides the values of the cost of using this algorithm extended to obstacle avoidance by AIVs (here for four obstacles of sizes: s_1 , s_2 , s_3 , and s_4). It appears that only the AIV_1 which circulates at the highest speed is impacted: the cost goes from 31% with the anti-collision algorithm alone to 44% with the extension for obstacle avoidance. For the other three AIVs, this extension of the algorithm to obstacle avoidance has no impact.

AIV	Velocity	3 obstacles sizes [1,2,3]	3 obstacles sizes [4,3,4]	4 obstacles sizes [1,2,3,4]	5 obstacles sizes [1,2,3,4,1]	10 obstacles sizes [1,2,3,4,1,2,2,4,1,3]
AIV_1	5	23	19	22	22	18
AIV_2	4	11	10	11	11	10
AIV_3	4	11	10	11	11	10
AIV_4	3	7	7	7	7	7

Table 2.2 – Sc_2 : obstacles avoidance algorithm simulation results

AIV	Velocity	Number of turns / 15mn nominal conditions	Number of turns / 15mn anti-collision algorithm	Anti-collision algorithm cost
AIV_1	5	39	22	44%
AIV_2	4	13	11	15%
AIV_3	4	13	11	15%
AIV_4	3	8	7	13%

Table 2.3 – Sc_2 : anti-collision and obstacles avoidance algorithm cost

We verify in these visual scenarios that the obstacles and other AIV agents are perceived by each AIV agent, and consequently, they will be able to avoid collisions. Therefore, the simulation validates the extended BAHNES’s algorithm with collision avoidance and fixed or dynamic obstacle detection processing.

2.2.5 Discussion

Beyond the simulation context, if we wish to cross the threshold of experimentation based on actual and cooperative mobile robots, we must take an interest in communication standards, starting with those presented in the **section 1.2.5**, in the **Table 1.2**. They are reported in the **Figure 2.2**.

It is then a question of whether it is possible to adopt the same standards in the industry as the one developed in the C-ITS community for road vehicles (or even adapt

the standards if it is more relevant).

In order for each vehicle in a fleet of vehicles to cooperate and provide relevant information to the other vehicles in the fleet, it is necessary that it can locate itself accurately. Using this condition as a springboard, we started by working on this aspect and have proposed a method.

We will therefore develop the problem of standardized messages in the rest of this section. The standardized cooperative messages defined by ETSI are of several types and have been presented in the **section 1.2.5**. We have selected the messages resumed in the **Table 1.2**: CAM, DENM, CPM and MCM.

The Bahnes' algorithm and our augmented Bahnes algorithm can then be transformed by replacing the messages indicated in **Table 2.4**:

Messages used in the augmented Bahnes' algorithm	ETSI messages
Hello_msg	CAM
Obstacle_msg	CPM
Alert_msg	DENM
Coop_msg / ACK_msg	no equivalent yet in the ITS standards, but will hopefull be replaced by future MCM or an extension of them

Table 2.4 – Correspondence between Bahnes' algorithm messages and ETSI messages

To explain the use of standardized ITS messages in our industrial context, we will consider the situation described in the **Figure 2.6**. This figure represents a complex intersection where multiple AIVs (AIV_1 , AIV_2 , and AIV_3) are navigating their paths. The AIVs need to communicate and cooperate effectively to avoid collisions, especially when crossing each other's paths. In this context, standardized cooperative messages as defined by ETSI play a crucial role in ensuring safe and efficient navigation.

Tag_i and Tag_j represent key decision points at the entrance and exit of the intersection, respectively. As the AIVs approach these tags, they must make informed decisions regarding whether to yield or proceed, based on the exchanged information with other AIVs in proximity.

Upon approaching the intersection, AIV_1 and AIV_2 must coordinate to determine the right-of-way. This coordination could involve messages that currently lack a direct equivalent within the ETSI standards but could potentially be represented by future MCM or an extension thereof.

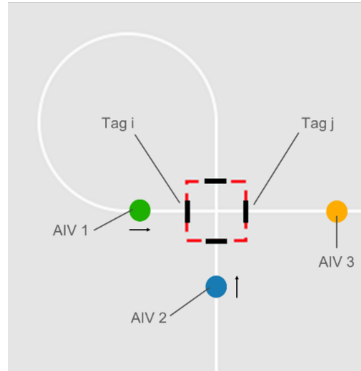


Figure 2.6 – Situation of intersection crossing by three AIVs

Figure 2.7 provides further detail on the types and sequence of standardized messages transmitted by AIV_1 as it navigates the intersection depicted in **Figure 2.6**.

Indeed, following the perception of the Tag_i announcing the intersection, AIV_1 must send a MCM that will be broadcast in the intersection area. Depending on the MCM received in return (from AIV_2 and AIV_3 , for instance), AIV_1 may decide to cross the intersection or wait until the intersection is cleared. While advancing in the intersection, AIV_1 will perceive the Tag_j indicating to it that the intersection is crossed. At this time, AIV_1 will send a MCM to signal any AIVs on standby that it is leaving the intersection area (AIV_2 , for instance).

2.3 Global obstacle avoidance by AIV with a collective strategy

In this section, we will propose a global obstacle avoidance with shared perception. Indeed, we will not use a local avoidance solution to improve autonomy. But rather a collaboration between the different vehicles of the fleet. The proposed approach is adaptable and flexible for centralized or decentralized solutions. In other words, we are dealing with a distributed system where each AIV decides by itself, augmented with communication between vehicles to share information about the environment. The communication may also go through a central supervisor before being redistributed.

The work of this section is outlined in [ASPAI22].

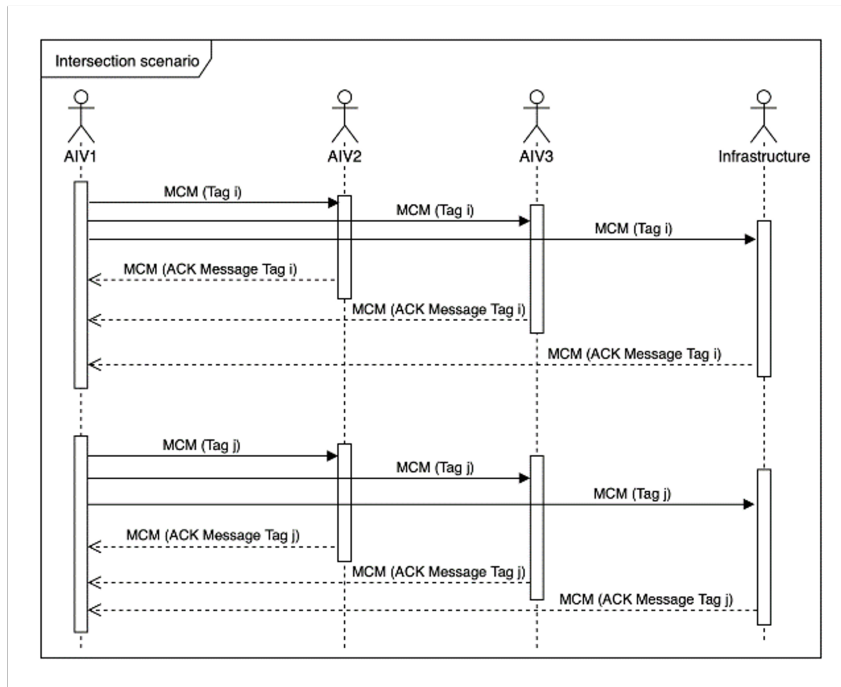


Figure 2.7 – Cooperative messages used for intersection crossing. (note: In a purely distributed scenario, infrastructure is not required)

2.3.1 Obstacle avoidance approaches: scenarios

We propose to study three approaches to obstacle avoidance by AIVs that will be tested in the simulation. Each robot has knowledge of the environment, i.e., the position of aisles, intersections, AIV stations, and mission points. These various important traffic points of the circuit will be called nodes in the following, and are identified by respective numbers shown in **Figure 2.8**.

In the first approach (*Strategy1*), agents do not have the ability to re-plan a route and change their route when faced with an obstacle.

In the second approach (*Strategy2*), AIV can change routes and plan a new route when an obstacle is detected, but they do not cooperate with other AIVs. This means that vehicles will enter routes where obstacles are present and will have to change their route, even though another vehicle has already perceived this. They are able to calculate the path to their destination.

The path-planning algorithm chosen for these experiments is Dijkstra. The objective is not to optimize a path planning algorithm (each AIV agent having to execute the same

algorithm) but to use an efficient algorithm to measure the performance of a collective strategy compared to an individual approach in carrying out missions in an industrial environment.

The last approach (*Strategy3*) will be a collective strategy upon detection of an obstacle. This strategy is established according to two points of view:

- The vehicle detects an obstacle, which determines the level of obstruction. Then, it communicates the description (position, level of obstruction, type – static or dynamic) of the obstacle to the other vehicles. It can react according to its static or dynamic type to avoid it if possible, or it can replan another path to reach its objective and accomplish its mission. This process is detailed in **Algorithm 1**.
- The vehicle receives information about an obstacle in the environment. It first tries to determine whether the obstacle is on its path to accomplishing its mission. If this is the case and the obstacle is obstructing its planned path, it can replan its route. If the obstacle can be avoided, it can act and anticipate this static or dynamic obstacle by arriving in the vicinity of the received position of the obstacle. This improves the safety of the area and prevents accidents. This step is described in **Algorithm 2**.

Algorithm 1: AIV detects an obstacle ahead

Input : AIV attributes, lidar function, communication

```

1 if AIV detects an obstacle then
2   | Determine level of obstruction
3   | Send message to other AIVs (obstacle, position, obstruction level)
4   | if Obstacle is bypassable then
5   |   | Bypass obstacle (static or dynamic)
6   | else
7   |   | if Obstacle is static then
8   |   |   | Recalculate path to continue its mission (e.g., Dijkstra algorithm)
9   |   | else
10  |   |   | Slow down and increase caution (dynamic obstacle: pedestrian, etc.)
11  |   |   | Recalculate path to continue its mission (e.g., Dijkstra algorithm)
12  |   | end
13  | end
14 end

```

Algorithm 2: AIV receives an information about an obstacle

Input : AIV attributes, lidar function, communication with other AIVs

```
1 if Obstacle is on its path then
2   | if Obstacle is bypassable then
3   |   | Bypass obstacle (static or dynamic)
4   | else
5   |   | if Obstacle is static then
6   |   |   | Recalculate path to continue its mission (e.g., Dijkstra algorithm)
7   |   |   | else
8   |   |   |   | Slow down, increase caution (dynamic obstacle: pedestrian, etc.)
9   |   |   |   | Recalculate path to continue its mission (e.g., Dijkstra algorithm)
10  |   |   | end
11  |   | end
12 end
13 Continue its path and its mission
```

2.3.2 Environment simulation and setting up the experiments

The environment chosen to test the different approaches is the warehouse presented by [TBS18], referenced as *IndustrialCircuit* in **Figure 2.2**. This specific circuit includes several intersections, where vehicles can arrive from different sides like in a warehouse. Thus, this kind of circulation plan provides the different characteristics of an industrial environment allowing us to conduct simulated experimental tests in accordance with realistic scenarios of an industrial context.

The directions of circulation have been adapted for our experiments and are identified in **Figure 2.8**. Five AIV agents are integrated into this environment corresponding to the five parking spaces available in this environment. AIVs agents must perform missions in the environment shown in **Figure 2.8**. A mission consists of travelling from its car park to a storage point and back to its car park.

Thus, the AIVs agents applying the collective intelligent strategy has two route calculations to perform with the Dijkstra algorithm. The first one is to calculate the shortest path to the storage point. The second one is to calculate the shortest route back to their assigned parking space.

The costs in distance between the different nodes (denoted as n_i) of the circuit have been chosen and applied to favour certain directions of traffic. For example, vehicles should go from n_{14} to n_6 , which has a cost of 10, rather than to n_5 , which has a cost of

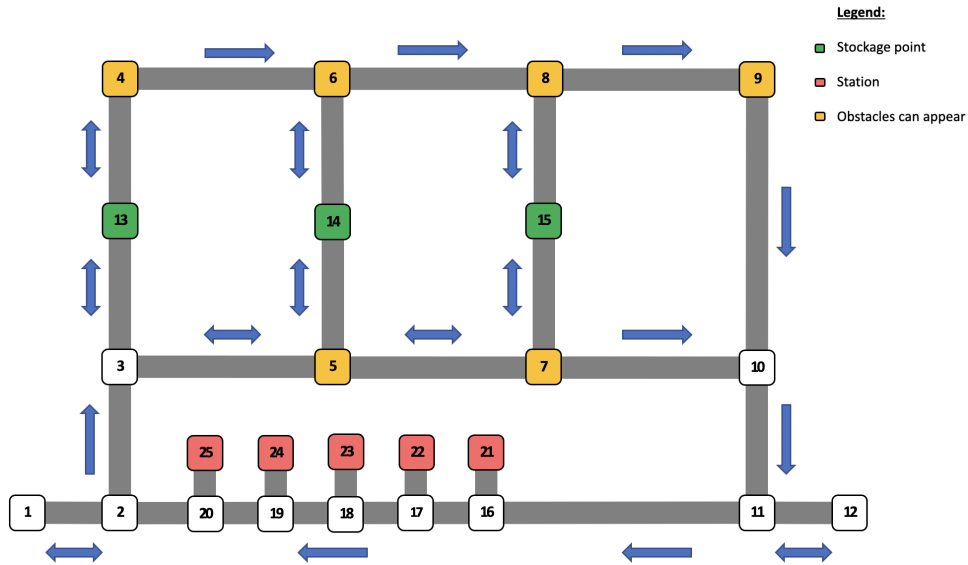


Figure 2.8 – *IndustrialCircuit* diagram

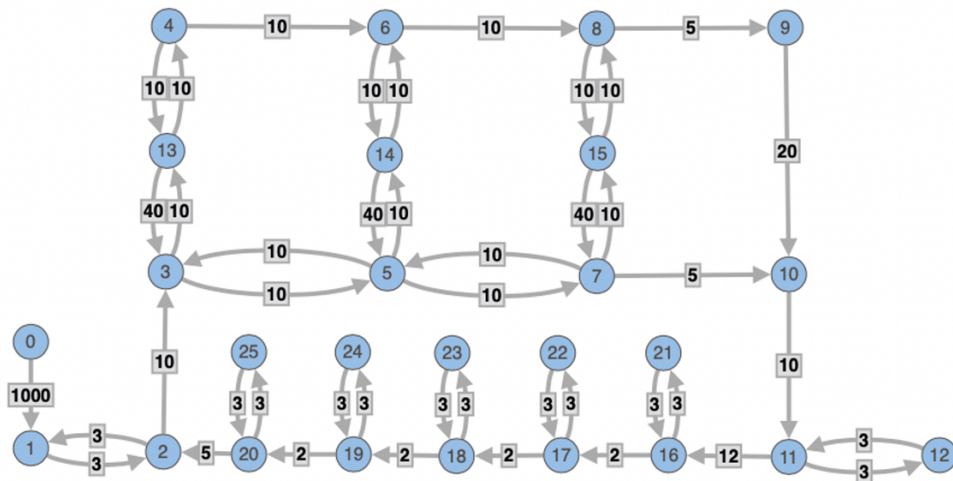


Figure 2.9 – Oriented graph of the *IndustrialCircuit*

40. These costs between the different nodes are shown in the directed graph in **Figure 2.9**. These costs are used by Dijkstra’s algorithm to find the path that costs the least distance, and therefore the fastest.

To simulate these different experiments, we implemented a graphical application with different functionalities. The different classes presented in **Figure 2.2** have been implemented in Python from scratch. Its interface is shown in **Figure 2.10**. In the central

part, we have the representation of the warehouse presented by [TBS18], which we have adapted in **Figure 2.8**. The different nodes present in **Figure 2.9** and **Figure 2.10** are represented by white squares in the interface. The vehicles are visualised by small colored circles, and the obstacles are colored black with different sizes corresponding to the obstruction levels in the aisle.

The functionalities proposed by our application (**Figure 2.10**) are described in **appendix B - section B.3**. For instance, in **Figure 2.10**, an avoidable obstacle with a size s_2 in n_6 is present, and an obstacle obstructs the whole passage with a size s_4 in n_8 . The size of the obstacle in the interface corresponds to its obstruction level.

These different approaches were detailed in **subsection 2.3.1**. *Strategy1*, therefore, does not allow the agents to use Dijkstra’s algorithm. *Strategy2* does, and *Strategy3* allows agents to communicate with each other about the description of a perceived obstacle.

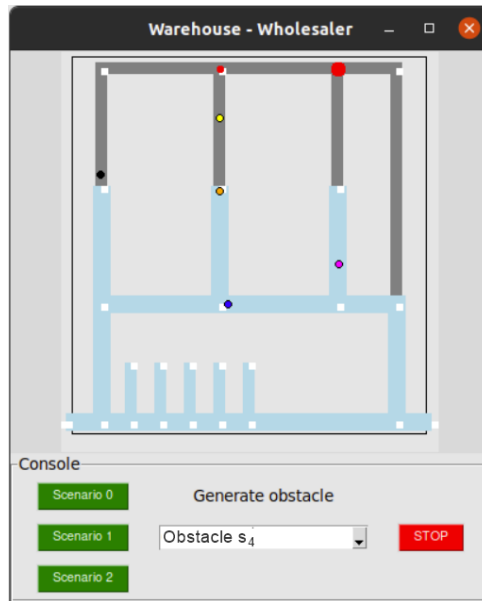


Figure 2.10 – Application interface to simulate missions in the warehouse environment

The three approaches were tested in six different experiments. Each experiment corresponded to an obstacle completely obstructing an aisle on one of the six nodes: n_4 , n_5 , n_6 , n_7 , n_8 and n_9 shown in orange in **Figure 2.8**. During these six experiments, the AIV agents had to perform one of three missions consisting of going from the car park to one of the nodes n_{13} , n_{14} and n_{15} randomly chosen by the supervisor, and then

returning to their parking place. They were equipped with one of the three approach capabilities described in **subsection 2.3.3**. That is, three experiments were performed with *Strategy1*, *Strategy2* and *Strategy3* with an obstruction at the same node n_i to compare the approaches.

The performance indicators that were used to define the efficiency and performance of these approaches are:

1. The successful completion of all missions assigned to each agent. This parameter *allMissionsCompleted* is *True* if all agents have completed all their assigned missions, and *False* if any missions remain incomplete. It is defined as in **Equation (2.1)**.

$$allMissionsCompleted = \begin{cases} \text{True} & \text{if } nbCompletedMissions = totalMissionsAssigned \\ \text{False} & \text{otherwise} \end{cases} \quad (2.1)$$

2. The cumulative distance travelled by all agents to complete the assigned missions: *totalDistanceTravelled*. It is expressed as in **Equation (2.2)** for five agents, where:

- N is the number of agents (in this context, $N = 5$).
- $nbMissions_i$ is the number of missions assigned to agent i .
- d_{ij} is the distance travelled by agent i to complete mission j .

$$totalDistanceTravelled = \sum_{i=1}^N \sum_{j=1}^{nbMissions_i} d_{ij} \quad (2.2)$$

3. The sum of costs - the overall time is taken to complete the various tasks: *TotalCompletionTime* (see **Equation (1.1)** formalised in **subsection 1.2.4**).

Distance and time are performance indicators that were chosen because they are strongly related to the energy used by an AIV. Thus, this indicates the energy impact of the different approaches.

2.3.3 Experiments results

Table 2.5 summarises the different results of the agents in the six experiments according to *Strategy1*, *Strategy2* or *Strategy3*. *Strategy1* does not allow the agents to complete all the missions, which is explained by the fact that they are not able to recalcul-

late their routes. Thus, if any agent was blocked by an obstruction on a node, it remained without the ability to react and the agents' overall mission failed.

<i>allMissionsCompleted</i>						
Strategy number / Obstacle node	n_4	n_5	n_6	n_7	n_8	n_9
Strategy1	False	False	False	False	False	False
Strategy2	True	True	True	True	True	True
Strategy3	True	True	True	True	True	True

Table 2.5 – Completed missions in relation to the strategy number of agents

Conversely, *Strategy2* and *Strategy3* applied by the agents allowed in both cases to fulfil all the missions thanks to their ability to adapt and to plan a new route (with the Dijkstra algorithm) when encountering an obstacle obstructing a path. The distance and overall execution time to complete all the missions have therefore been quantified in **Table 2.6** and **Table 2.7** respectively.

<i>totalDistanceTravelled</i>						
Strategy number / Obstacle node	n_4	n_5	n_6	n_7	n_8	n_9
Strategy2	600	740	660	660	740	755
Strategy3	600	700	640	660	710	715

Table 2.6 – Total distance travelled by agents in relation to the strategy number

Total execution time: <i>TotalCompletionTime</i>						
Strategy number / Obstacle node	n_4	n_5	n_6	n_7	n_8	n_9
Strategy2	489,75	677,56	521,23	559,59	582,97	567,52
Strategy3	482,14	662,82	456,60	542,04	466,35	470,40

Table 2.7 – Total execution time to carry out the missions in relation to the strategy number

The analysis that emerges is that *Strategy3* allows the agents to travel less distance in 4 out of 6 experiments and allows them to complete the missions more quickly in all six experiments.

The cooperative perception allows the agents to not go directly to the obstacle if it is on their way, but to redefine with the help of Dijkstra another path to reach their

goal. Thus, logically, this collaboration takes precedence over the individual strategy, and communication in a fleet of AIVs thus increases their autonomy and their ability to adapt to new situations. Over the six experiments, there is a 3% gain in overall distance and a 9% gain in time for the whole fleet of AIVs. This cooperative perception is an essential basis for a future collective global strategy for the sharing and organisation of distributed missions within a fleet.

2.3.4 Discussion

The comparison of the three approaches evaluated using distance, execution time and whether or not a mission has been successfully achieved, showed that it is essential that vehicles have the ability to re-plan their routes while necessary. In addition, the last approach, which allowed agents to communicate the description of perceived obstacles, allowed for gains in distance and execution time of overall missions. Indeed, it allows AIVs to avoid traffic around an area where a static or dynamic obstacle (human operator, for instance) is present in an aisle. It also allows operators or other robots to be safe in that area: for example, to remove a static obstacle.

In the context of Industry 4.0, mobile robots need to become increasingly autonomous to intelligently avoid obstacles. We explored the potential impact of cooperative perception among the vehicles of a fleet during a mission. We aim to elevate collaboration to a higher level, allowing for mission sharing and reorganization in a distributed manner. We will see our proposition for this type of collective strategy in the **chapter 3**.

This study already shows the interest in collaboration to increase the collective and individual efficiency of the vehicles in a fleet. The perspectives that emerge from this contribution are to increase the work of the analysis with other performance indicators, but especially more experiments. Indeed, we can define several missions in a row by robots with several obstacles obstructing the lanes at different places.

2.4 Preparation of real experiments of an intersection scenario

In this section, our objective is to transition from simulated obstacle avoidance strategies to real-world experiments. Before proceeding to physical experiments, an advanced simulation step is necessary: emulating robots in Gazebo using ROS2. This step is crucial for validating our approach.

We aim to verify Bahnes’ algorithm utilizing ETSI messages, as discussed in **section 2.2.5** and particularly, shown in **Table 2.4**. CAM, DENM and CPM are important ETSI messages standardized. Therefore, we will propose a model of these messages adapted to the industrial context which has different constraints than the road sector, using the ROS2 framework. Following this, we will outline the experimental context and detail the emulation settings. Then, we will describe the communication setup between the robots. Furthermore, we will analyze the results within this emulation scenario. Finally, we will discuss the implications of MCM and their possible applications from an experimental perspective.

This section transcribes the work presented in [ARAC24].

2.4.1 Representation of ETSI messages for the industrial context

The purpose of *Hello_msg* proposed by Bahnes corresponds to CAM. This is a message sent by the vehicle to indicate its position in real-time. We propose a model for the industrial context in **Table 2.8** and its associated implementation as a message in ROS2 in **Table 2.9**.

CAM	ItsPduHeader		
	Cooperative Awareness Message	GenerationTime	
		CAM Parameters	BasicContainer (CurrentPosition + StationType)

Table 2.8 – Representation of CAM Message for the industrial context

CAM	ItsPduHeader its_header	uint8 protocol_version
		uint8 message_id CAM = 1 DENM = 2 CPM Message = 3 MCM = 4 ACK_MCM = 5
	uint32 station_id	
	uint16 generation_time	
StationType station_type	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3	
	float64[] current_position	

Table 2.9 – Modelization of CAM in ROS2

The alert message we proposed for the simulation of the augmented Bahnes algorithm will be implemented using the DENM. This can be of 3 different types: *TRIGGER*, *UPDATE* and *TERMINATE* (message_type in **Table 2.11**):

- *TRIGGER*: The initial alert message issued when an event is detected.
- *UPDATE*: A message used to update information related to the initial alert, providing additional or revised data.
- *TERMINATE*: A message indicating that the alert is no longer active. It contains the cause and sub-cause of the alert, which are specified in the *SituationContainer* block of the message (**Table 2.10**).

Several alert message codes have been transposed from the standard for our experiments such as *Collision_RISK* with sub-causes associated with this code such as a longitudinal, lateral, intersection-related or vulnerable user collision risk (modelling of *SituationContainer* in **Table 2.11**).

Augmented Bahnes algorithm allows vehicles to take obstacle detection into account. This augmentation has seen the arrival of a new message: *Obstacle_message* presented in **section 2.2**. CPM defined by ETSI has the same objective: inform vehicles in the same geographical area of the presence of pedestrians, obstacles, etc. The study [SBL19] reported the architecture of CPM. They are structured with a header containing the general information of a message, and a body specifying the information related to the detected element. The latter is divided into 3 parts:

DENM	ItsPduHeader		
	Decentralized Environmental Notification Message	GenerationTime	
		DENM Parameters	ManagementContainer (StationType + DetectionTime + Distance + ValidityDuration)
SituationContainer (CauseCode + SubCauseCode + InformationQuality)			

Table 2.10 – Representation of DENM for the industrial context

DENM	ItsPduHeader its_header	uint8 protocol_version	
		uint8 message_id CAM = 1 DENM = 2 CPM = 3 MCM = 4 ACK_MCM = 5	
	uint32 station_id		
	uint8 message_type TRIGGER = 1 UPDATE = 2 TERMINATE = 3		
	StationType station_type	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3	
	ManagementContainer management_container	uint64 detection_time	
		float64 distance	
		uint32 validity_duration (seconds)	
SituationContainer situation_container	CauseCode cause_code	uint8 value TRAFFIC_CONDITION = 1 ACCIDENT = 2 SLOW_AIV = 26 COLLISION_RISK = 97	
	CauseCode sub_cause_code	UNAVAILABLE = 0 LONGITUDINAL_COLLISION_RISK = 1 CROSSING_COLLISION_RISK = 2 LATERAL_COLLISION_RISK = 3 INVOLVING_VULNERABLE_USER = 4	
	uint8 information_quality UNAVAILABLE = 0 LOWEST = 1 HIGHEST = 7		

Table 2.11 – Modelization of DENM in ROS2

- **station data and CPM management:** information on the ITS station transmitting the message (position, heading)
- **sensor information:** details of on-board sensors (number, range, aperture angles)
- **perceived objects:** list of relevant objects detected (relative position and speed, dimensions, etc.)

Our CPM-inspired adaptation for the industrial context is shown in **table 2.12** and our ROS2 implementation is modelled in **table 2.13**.

CPM	ItsPduHeader			
	Collective Perception Message	GenerationTime		
		CPM Parameters	BasicContainer (CurrentPosition + StationType)	
			SensorInformationContainer (type + confidence)	
			PerceivedObjectContainer (objectID + distance + acceleration + yawAngle)	
NumberOfPerceivedObjects				

Table 2.12 – Representation of CPM for the industrial context

2.4.2 Experimentations context and settings

In our experimental context, we use four Turtlebot3 "burger" robot named $robot_1$, $robot_2$, $robot_3$ and $robot_4$, as our representation of AIVs. These robots are equipped with different components as described in **Figure 2.11** and use a Raspberry Pi and the ROS2 framework.

Before simulating the Bahnes' algorithm or other cooperative algorithms with standardized messages with the ROS2 framework and Gazebo, we need to set up the task assignment for AIVs to implement robot control and movement utility related to the notion of tasks/missions. We have assumed that the tasks are known and therefore configured in a file with the destination of each robot for each task. The destinations where the robots must perform tasks are represented by *PositionAction* which is (x, y) coordinates relative to the 2D world simulated in Gazebo. Our robotic architecture is similar to that used by [TCN21], a ROS2 toolbox for cooperative robotics.

That is, we have a layer to guide and a layer to plan the destinations of the robots. Indeed, each robot is associated with a *planner_client* node which sends the *PositionGoal* related to the destination of the task associated with the robot. Then, the *planner_server* node allows us to standardize this destination point which it publishes to the goal topic

CPM	ItsPduHeader its_header	uint8 protocol_version
		uint8 message_id CAM = 1 DENM = 2 CPM = 3 MCM = 4 ACK_MCM = 5
		uint32 station_id
	uint16 generation_time	
	StationType station_type	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3
	float64[] current_position	
	SensorInformation sensor_information	uint8 type UNKNOWN = 0 LiDAR = 1
		uint8 confidence UNKNOWN = 0 LOW = 1 MEDIUM = 2 HIGH = 3
	PerceiveObjectContainer perceive_object	uint8 objectID UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 OBJECT = 3
		float64[] distance
float64[] acceleration		
float64[] yaw_angle		

Table 2.13 – Modelization of CPM in ROS2

associated with the robot. Afterwards, the guidance layer will subscribe to this topic to control the robot until it reaches its mission position. Once the robot has reached its destination, the odometry server sends feedback to the *planner_client*, which can reply to the next *PositionAction* to the *planner_server*.

The communication between the different nodes for the control of *robot₁* for example is shown in **Figure 2.12**. Once the tasks have been assigned and the robots have been controlled, the robots must be able to communicate. This will allow the exchange of

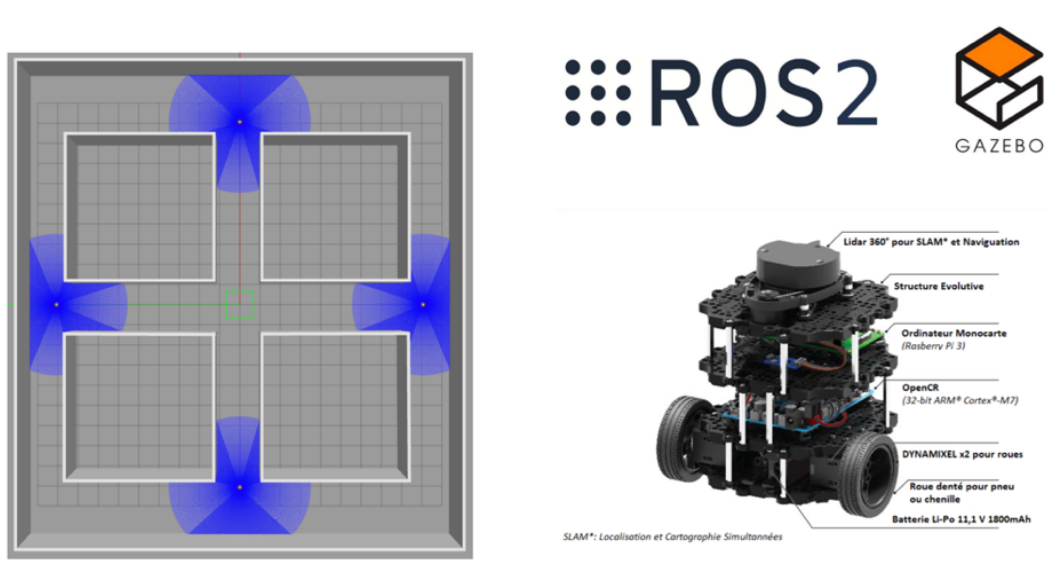


Figure 2.11 – Gazebo Simulation environment for experimentations

information on their positions, their vision of the environment, or their trajectory intention in order not to collide with each other or with obstacles.

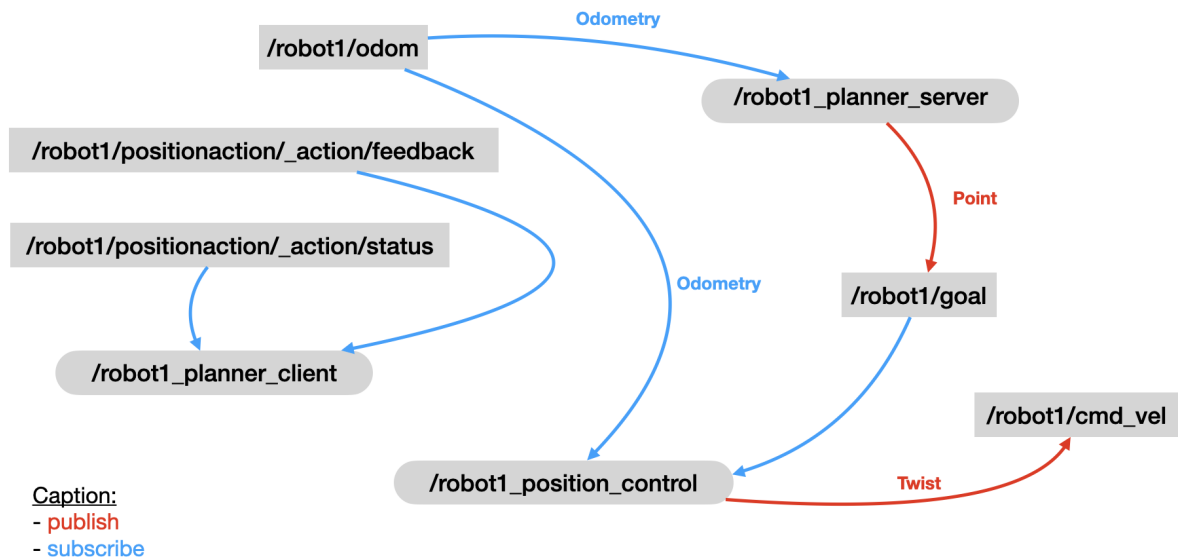


Figure 2.12 – Nodes and topics representation for *robot1* for path planning

2.4.3 Communication

In order to implement algorithms for collective cooperation between autonomous vehicles, we have implemented the different standardized messages presented in **section 2.4.1**. The implementation of those messages in ROS2 will allow robots to exchange these types of messages through topics.

The `turtlebot3_position_control` node of a robot allows it to control its speed and orientation towards its destination point defined by the `/goal` topic as explained in **section 2.4.2**. This node will also allow to exchange messages related to its observations of the environment thanks to its LiDAR. We have defined an observation distance and a safety distance. When the LiDAR detects something corresponding to the observation distance or less it publishes a CPM message in the topic corresponding to the robot with the related information. Similarly, if it detects something at a distance less than its safety distance, it sends an alert message, i.e., a DENM with the information associated with the DENM topic. This process of subscribing and publishing messages similar to all robots is modelled in **Figure 2.13** using `robot1` as an example.

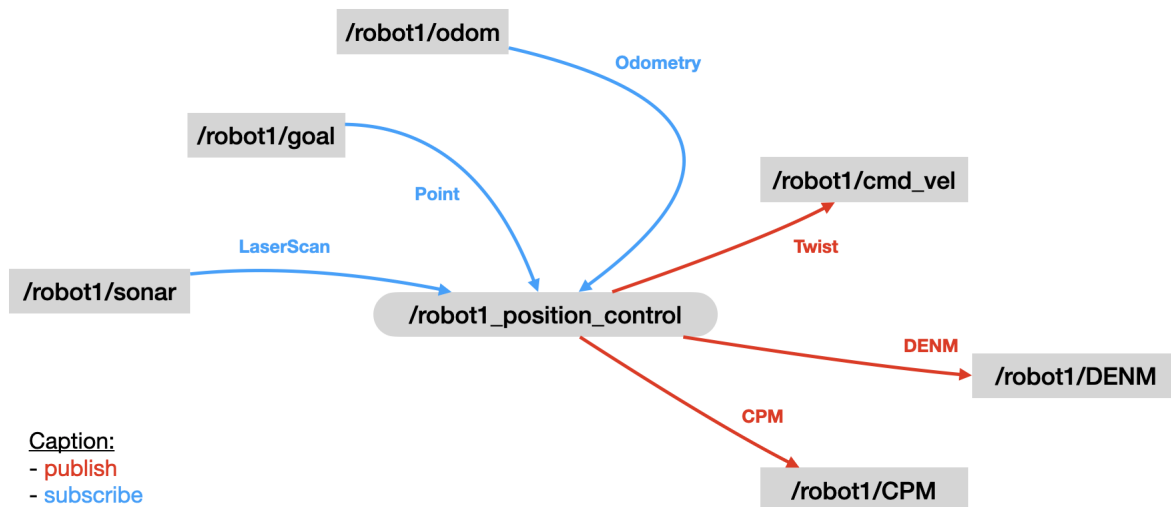


Figure 2.13 – Process of publishing/subscribing topics for `position_control` node of each robot

CAM are published in CAM topics similar to CPM and DENM. They are standardized using information from the robot's odometry sensor. This information published by one robot is retrieved by all other robots in the manner of a broadcast exchange. The architecture has been implemented so that each robot has an `exchange_messages` node

which allows it to subscribe to all the message topics of the other robots, i.e., CAM, CPM and DENM in our current experiment. These exchanges are modelled in **Figure 2.14** for two robots, but the process is similar regardless of the number of robots.

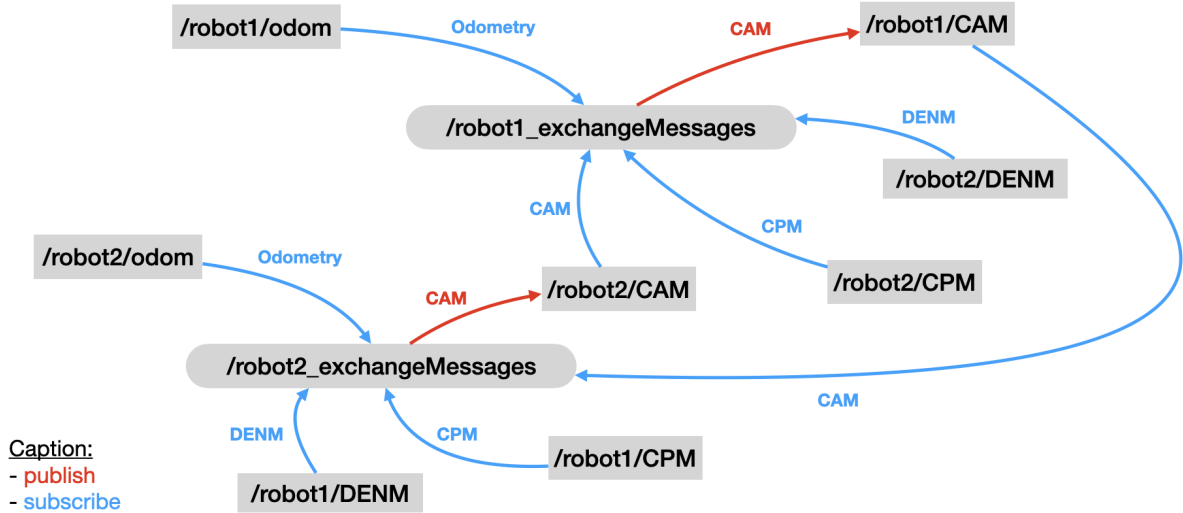


Figure 2.14 – Nodes and topics representation for exchanging messages

2.4.4 Results

Before actually experimenting with the strategic cooperation of the robots to avoid real collisions, we tested a simple scenario in a Gazebo simulation. Each robot was given the task of crossing the intersection shown in **Figure 2.11**, and they were asked to go to the *PositionGoal* in front of them.

In this intersection scenario, the robots broadcast their positions using CAM while moving towards the intersection. The LiDAR of *robot₂* detects an obstacle and sends a CPM to share its information. But as we have not implemented obstacle avoidance control, it will stop at the safety distance of the LiDAR and send a DENM with a risk of longitudinal collision. Similarly, *robot₃* and *robot₄* did not encounter any problems on their paths but when crossing the intersection, their LiDAR detects at a safe distance the other robot with an angle of less than 45°. They will immediately send a DENM of risk of longitudinal collision as well.

It can be noted that unlike *robot₂*, *robot₃* and *robot₄* did not send a CPM before, because, at the moment of the intersection, they were already too close. *robot₃* and *robot₄*

stop to avoid collision. Our goal was to find a cooperative solution for AIVs to avoid collisions and obstacles, but also to avoid unnecessary braking and stopping to optimize energy and speed. As a first step, these results show that our cooperative strategy allows the robots not to collide with each other or with obstacles.

Our results of communication between the different robots are therefore validated in the Gazebo simulation environment. Nevertheless, in order to be able to find a cooperation to cross the intersection it would be necessary to implement the cooperative messages presented by the augmented Bahnes algorithm.

2.4.5 Discussion

For this purpose, we previously discussed a service not yet implemented by ITS WG1 and ETSI: MCM. We then propose in the same way a representation for the industrial context in **table 2.14** and a modelling of the message for ROS2 in **table 2.15**. This message would share the information that one wishes to cross an intersection by indicating *ManeuverContainer* information, that is id of the intersection, as well as the direction one would take.

Thus, if we assume that the vehicles know the positions of the different intersections, or are able to locate them, can send an MCM to the other robots indicating their planned trajectory in the intersection. The robots concerned by the request will be able to respond to an *ACK_Message* indicating their agreement or disagreement with the request. The industrial representation of this message and the ROS2 modelling we propose are detailed in **tables 2.16 and 2.17**.

This discussion around MCM to enable vehicle cooperation when approaching an intersection raises an issue: if several vehicles request to cross the same intersection at the same time, or if one vehicle indicates that it does not agree to the request of another vehicle, a deadlock situation arises. There are three possibilities to manage this concern:

- The first is the idea of **strong cooperation**: characterized by agreement-based cooperation. Here, all vehicles automatically agree with one another's requests, ensuring that no conflicts arise and coordination is seamless. This method relies on a predefined consensus that avoids the need for negotiation.
- The second is to **set up a supervisor** who can arbitrate in situations of conflict. This involves centralized arbitration, where an external entity decides which vehicle should proceed, thereby resolving the deadlock. However, this method moves away from a fully distributed architecture, as it depends on a central decision-maker.

- Finally, the last solution is to incorporate an **algorithmic layer** that is known to all the robots, and which therefore serves as a decisive judgment. This solution represents negotiation-based cooperation with algorithmic resolution. In situations of disagreement, the algorithm would evaluate factors such as task priority and urgency, guiding all vehicles to reach the same decision autonomously. For example, if one robot disagrees with the crossing of another, it is then the level of priority and urgency of the task between the two robots that will decide who will be the first to cross. This maintains the principle of strong collaboration while allowing the fleet to resolve conflicts through a shared, systematic process.

These solutions, related to how MCM can be utilized to handle conflicts at intersections, align with **Level 3** of autonomy from a collective perspective for a fleet of AIVs (as detailed in **chapter 1 - subsection 1.3.3**). At this level, vehicles engage in collective decision-making and task allocation, using communication protocols such as V2X messages (including MCM). The vehicles make coordinated decisions on maneuvers based on shared priorities and urgency, ensuring effective and coordinated actions across the fleet.

MCM	ItsPduHeader			
	Maneuver Coordination Message	GenerationTime		
		MCM Parameters	BasicContainer (CurrentPosition + StationType)	
			ManeuverContainer (ReferenceIntersection + Direction)	

Table 2.14 – Representation of MCM for the industrial context

MCM	ItsPduHeader its_header	uint8 protocol_version uint8 message_id CAM = 1 DENM = 2 CPM = 3 MCM = 4 ACK_MCM = 5 uint32 station_id
	uint16 generation_time	
	StationType station_type	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3
	float64[] current_position	
	ManeuverContainer maneuver	uint8 id_intersection uint8 direction STRAIGHT = 0 LEFT = 1 RIGHT = 2

Table 2.15 – Modelization of MCM in ROS2

ACK_MCM	ItsPduHeader			
	Acknowledgement Maneuver Coordination Message	GenerationTime		
		ACK_MCM Parameters	BasicContainer (CurrentPosition + StationType)	
			DestinaterContainer (StationType + StationID)	
			ManeuverContainer (ReferenceIntersection + Direction)	
AckResponse				

Table 2.16 – Representation of ACK_MCM for the industrial context

ACK_MCM	ItsPduHeader its_header	uint8 protocol_version
		uint8 message_id CAM = 1 DENM = 2 CPM = 3 MCM = 4 ACK_MCM = 5
		uint32 station_id
	uint16 generation_time	
	StationType station_type	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3
	float64[] current_position	
	StationType station_type_destinater	uint8 value UNKNOWN = 0 PEDESTRIAN = 1 AIV = 2 BEACON = 3
	uint32 station_id_destinater	
	ManeuverContainer maneuver	uint8 id_intersection
		uint8 direction STRAIGHT = 0 LEFT = 1 RIGHT = 2
bool ack_mcm_response		

Table 2.17 – Modelization of ACK_MCM in ROS2

2.5 Conclusion

In an Industry 4.0 context, many actors cross paths in different areas of a warehouse or a factory: vehicles, operators and obstacles (objects that fall or are left in the aisles may appear). Obstacle avoidance which is part of the navigation process of robots can be improved by collective strategies.

A specific state-of-the-art on used message-based communication protocol between vehicles to prioritise the passage through an intersection allowed us to identify Bahnes' algorithm [BKH16], well representative of the cooperative strategies developed in the field.

Developing the autonomy of AIVs requires a relevant working method. The identification of reusable or adaptable algorithms to the various problems raised by the increase in the autonomy of AIVs is not sufficient, it is also necessary to be able to model, simulate, test and experiment with the proposed solutions. Simulation is essential since it allows both to adapt and validate the algorithms, but also to design and prepare the experiments.

In this chapter, we introduced an improved version of BAHNES' algorithm to handle both stationary and dynamic obstacles. Then, we proposed an ABMS to help develop and test scenarios simulation. We conducted a first experiment on the "4-Lap Circuit" to verify the augmented BAHNES' algorithm before discussing the improvement of the message-based communication proposed with ETSI messages. The improved BAHNES' algorithm corresponds to "Level 2" on our proposed autonomy scale from a collective perspective for a fleet of AIVs (as detailed in **subsection 1.3.3**). This level is characterized by interoperability and cooperative awareness among heterogeneous robots. By enhancing obstacle detection and avoidance, this contribution advances the capabilities of AIVs to share information about obstacles and therefore help to create dynamic environmental maps, which is essential for seamless operation in a collective context.

Afterwards, we proposed a global obstacle avoidance strategy using shared perception also falls under "Level 2" on the collective autonomy scale. We showed with three different approaches, the impact of the cooperation between AIVs. Accurate, real-time position estimation of AIVs is crucial for these strategies to work, as it helps each vehicle make better decisions and improves the overall system. To address this, we propose in **appendix C**, a method for estimating AIV positions, as accurate positioning is crucial for the success of these collective strategies.

Then, we adapted certain ETSI standard messages (CAM, DENM, and CPM) for the Industry 4.0 context, which are crucial for achieving "Level 3" on the collective autonomy scale. We tested these communication protocols using Turtlebot3 "burger" robots in the ROS2 framework and simulated a simple intersection scenario in Gazebo. By introducing and simulating MCM, ACK_MCM, and other messages in this context, our work enables AIVs to engage in collective decision-making and task allocation. This represents a significant advancement towards higher autonomy, where the fleet can collaboratively plan maneuvers and optimize their actions.

As an extension of this latter work and in order to perform real experiments, we discussed these results as well as a cooperation message named MCM. This implementation perspective would allow us to validate the augmented BAHNES algorithm with the exchange

of cooperation messages to describe one's intention to cross an intersection.

To conclude, this chapter opens the door to more advanced global collective strategies with the possibility of the allocation, scheduling and distribution of tasks between them in real-time after the perception of an obstacle. Furthermore, the tracking of obstacles by the robots, or assisted by the infrastructure, i.e. a cooperative perception, would also optimize the efficiency of the fleet in carrying out these missions.

COLLECTIVE TASK ALLOCATION STRATEGIES

3.1 Introduction

In the context of Industry 4.0, effectively managing fleets of AIVs is essential for enhancing productivity and addressing the complexities of contemporary industrial workflows. These AIVs are often required to execute tasks that are subject to specific time and priority constraints. To ensure that all tasks are completed effectively, it is crucial to develop and implement sophisticated Task Allocation (TA) strategies that enable seamless operations across the entire fleet.

Moreover, the autonomy of an AIV fleet can be further increased if the AIVs can collectively manage and optimize the TA problem: that means they collectively assign the set of tasks to the set of AIVs with a high level of efficiency (for instance, minimization of energy costs or mission completion time) [DVD20].

This chapter explores collective TA strategies, emphasizing the role of V2X cooperation in enhancing the coordination and communication among AIVs. The concept of “collective” in this context refers to the collaborative efforts of multiple AIVs working together towards common goals, such as completing tasks in an efficient and timely manner. By adopting a collective approach, AIVs can share information, dynamically adjust their actions and optimize task distribution across the fleet. This collective strategy is especially beneficial in complex industrial environments where tasks are interdependent, and the success of one AIV often impacts the performance of others.

Firstly, in **section 3.2**, we propose a dynamic TA strategy in V2X cooperation mode with the infrastructure. It lays the groundwork by introducing a formal TA model to facilitate collective TA among AIVs. It highlights the importance of effective communication and coordination, which are essential for AIVs to collaborate and share environmental information. Although existing ETSI messages support certain types of vehicular com-

munication, they do not cater to the collective TA processes required in Industry 4.0. To address this gap, we propose new standard message types that enable AIVs to cooperate more effectively in TA processes, ensuring that all tasks are completed according to their constraints.

Then, in **section 3.3**, we apply the proposed TA strategies to a practical scenario involving a fleet of homogeneous mobile robots operating in a warehouse environment. These robots are tasked with loading and unloading goods, and the simulation is designed to test the TA processes in dynamic, real-world conditions. Building on the ABMS framework from the previous chapter, this section verifies the effectiveness of the collective TA strategies by analyzing the performance of the fleet in various scenarios. The simulation results provide valuable insights into how AIVs can dynamically adjust their tasks and re-allocate them in response to changing conditions.

Finally, **section 3.3** addresses a critical challenge in collective TA: the potential presence of faulty agents within the infrastructure, such as an AIV, a camera transmitting incorrect messages due to sensor miscalibration or other issues. Specifically, we focus on the problem of a faulty camera within the infrastructure, which could lead to incorrect TA and disrupt warehouse operations. This section explores a solution designed to promptly detect and mitigate the impact of such faults in an example scenario, ensuring that the integrity and efficiency of the warehouse environment are maintained even in the presence of faulty equipment. By addressing these challenges, we aim to enhance the robustness and reliability of collective TA strategies in industrial settings.

In summary, this chapter aims to advance the understanding and implementation of collective TA strategies in industrial settings. By leveraging V2X cooperation and decentralized decision-making, the proposed strategies enable AIV fleets to operate more efficiently, adapt to dynamic environments, and achieve greater overall productivity.

This chapter is related to the work published in [ICAE24].

3.2 Dynamic TA strategy in V2X cooperation mode with the infrastructure

In the realm of Industry 4.0, the deployment of AIVs within a fleet is a pivotal component in optimizing operational efficiency and productivity. These AIVs are tasked with executing a variety of assignments that are defined by specific time and priority constraints. The primary objective of the fleet is to ensure the completion of all tasks while

adhering to these constraints, thereby ensuring seamless operations.

As previously mentioned, we will propose a formal TA model which will serve as a foundation throughout this chapter. However, in order to define a collective strategy for TA, effective communication is crucial for AIVs to coordinate their activities and share information about their environment. Some standard ETSI messages exist, but no one for the collective TA process. This seems logical because in the road field, AV do not have a common objective as in Industry 4.0. Consequently, in **subsection 3.2.1**, we will propose two types of standard messages to cooperate between them for TA process. Then, we will define in **subsection 3.2.3** a proposed solution for the tasking process in the way of decentralization for AIV with a collective strategy.

3.2.1 Formal TA problem

The common objective of the AIVs belonging to the same fleet is to perform all the tasks assigned to them while respecting a certain number of time and priority constraints. In this context, given T a set of tasks to be performed, a task $\tau \in T$ is defined by the following tuple (**Equation (3.1)**):

$$\tau = \langle \tau_{id}, \kappa, p_{start}, p_{end}, t_{start}, t_{end}, tau_{pr}, \sigma \rangle \quad (3.1)$$

Where τ_{id} is the task identifier; κ is the task category (for example in a warehouse, move goods, i.e.: “load the goods at a starting point to bring it to an ending point”); p_{start} is the task starting point; p_{end} is the task ending point; t_{start} is the task starting time; t_{end} is the task ending time; tau_{pr} is the task priority; and σ is the task status.

A set of n tasks can be grouped to form a mission m defined as follows (**Equation (3.2)**):

$$m = \langle \tau_1, \dots, \tau_n \rangle \quad (3.2)$$

Given two sets V and T , such that $V = \{v_1, \dots, v_n\}$ is a set of n AIVs and $T = \{\tau_1, \dots, \tau_m\}$ is a set of m tasks, we define the two following functions:

- the function *COST* which allows to calculate the cost $c_{i,v_j} \in C$ of performing a task τ_i by an AIV v_j (**Equation (3.3)**);
- the function *ALLOCATION* which allows to allocate each task $\tau_i \in T$ to an AIV $v_j \in V$, depending on the cost c_{i,v_j} (**Equation (3.4)**).

$$COST : T, V \rightarrow C \quad (3.3)$$

$$ALLOCATION : T, V, C \rightarrow A \quad (3.4)$$

3.2.2 Communication

In order to successfully perform all assigned tasks, AIVs must coordinate and therefore cooperate and share information about their activity and their perceptions of the environment. In our works in the **chapter 2**, AIVs used ETSI messages presented in the **Table 1.2**. Indeed, AIVs had the possibility to communicate about their localization with CAM and communicate about perceived obstacles with CPM and DENM to prevent an unexpected event.

Another type of V2V communication could be useful to improve cooperation between AIVs in carrying out their tasks. Indeed, if an AIV finds itself blocked by obstacles, broken down or generally unable to perform the task in progress, it sends a DENM by default. However, It could then be useful for it to send a cooperative message to delegate the realization of his task with the necessary information.

We therefore propose a new Cooperative Task Message (CTM), which would allow in particular delegating a task. [Haf+13] propose a protocol with four new types of messages, including the Cooperative Response Message (CRM) for transmitting the response to a request for cooperation. The AIV agents, modelled in the simulation, will use this type of message in feedback from the CTM to signify their agreement to take charge of a task for example.

3.2.3 Tasking process

We will define a TA process based on a market model type solution [HK13]. The flexibility of this solution allows a good adaptation for a decentralized system. Its process is depicted in **Figure 3.1** including these following key moments:

- task definition and assignment
- task clustering
- auction process
- optimization and task (re-)allocation

Task definition and assignment

Firstly, tasks are usually defined by an organizational actor known as the supervisor, as illustrated in *step 1* of the **Figure 3.1**. The supervisor sent using a CTM to an available AIV (for instance, having no tasks to be accomplished at this time). This AIV will act in the role of auctioneer. Upon receiving the task package, this AIV auctioneer sends an acknowledgement using a CRM, completing *step 2*.

Task clustering

For greater efficiency, before the auction process begins, the AIV auctioneer can cluster certain tasks received, as outlined in *step 2*. This involves, in particular, associating tasks that have common starting or ending points to increase efficiency.

In the context of a warehouse environment, where logistics operations often involve moving materials between different locations, the clustering of tasks can significantly enhance efficiency. Warehouses typically have various nodes, such as parking sources, storage points, and distribution areas, where materials are frequently moved. By strategically grouping tasks that share common starting or ending points, the AIV auctioneer can minimize unnecessary movements and streamline the overall workflow.

For example, consider the task ($\tau_1 = [21, 13]$) means: *"Bring material from a parking source node n°21, to a storage point node n°13"*. An associated task τ_2 could be to go to a parking lot after completing task τ_1 , for instance: *"Take material from storage point node n°13 and bring it to parking node n°25"* ($\tau_2 = [13, 25]$). Then, the two clustered tasks τ_1 and τ_2 are represented by the mission $m_i = \langle \tau_1, \tau_2 \rangle = \langle [21, 13], [13, 25] \rangle$.

In the tables of **section 3.3.4**, we simplify the writing of the mission m_i by using the triplet $\langle 21, 13, 25 \rangle$.

Auction process

Once tasks are clustered into missions, the AIV auctioneer sent these missions to all AIVs (*step 3*). Each AIV calculates the costs associated with executing the various missions, considering a range of performance indicators such as distance, energy consumption, and time. These calculations inform their bids, which are then submitted back to the auctioneer.

Optimization and task (re-)allocation

Each AIV returns all of its bids to the AIV auctioneer. Upon receiving the bids, the AIV auctioneer performs a simple optimization algorithm (*step 4*). The algorithm prioritizes clustered tasks by listing them in order of priority. Subsequently, the AIV that submits the lowest-cost bid for a mission awarded the task, following a strategy that prioritizes cost-effectiveness.

To summarise the communication process, the AIV auctioneer allocates tasks (eventually clustered tasks) to each selected AIV, sending to it via a CTM (*step 5*). The AIV confirms receipt of the CTM by sending back a CRM to the auctioneer, completing *step 6*. The different interactions between the supervisor, the AIV auctioneer and the other AIVs are depicted in the sequence diagram of **Figure 3.2**.

This allocation mechanism also supports task reallocation. An auctioneer can himself become an auctioneer to re-auction a task (by following *steps 2 to 5*) if necessary, by managing the redistribution of all or part of his tasks. The AIV with the best bid in this re-auction will then add the reallocated task(s) to its assigned set of tasks to complete.

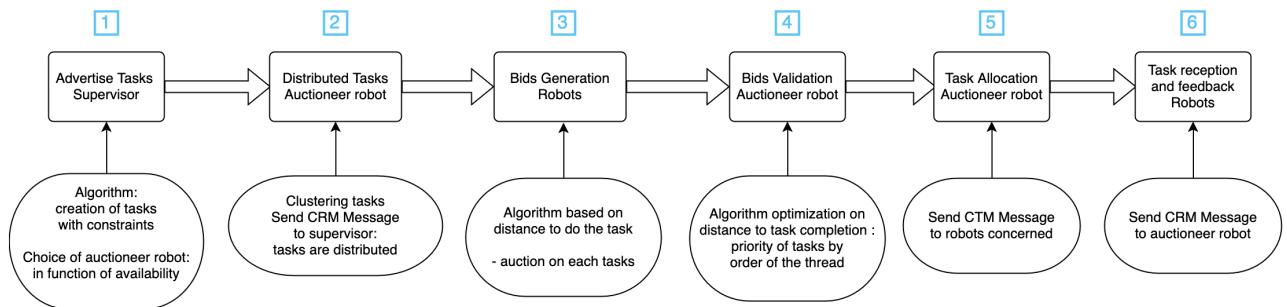


Figure 3.1 – Task allocation process

3.3 Dynamic task (re-)allocation simulation

The objective of this section is to apply and test the TA process, as presented in the previous section, on a fleet of homogeneous mobile robots, loading and unloading goods in a warehouse. The methodology preceding the simulation involves augmenting the ABMS proposed in **chapter 2**. The aim is to fit the scenarios to be tested in order to verify the

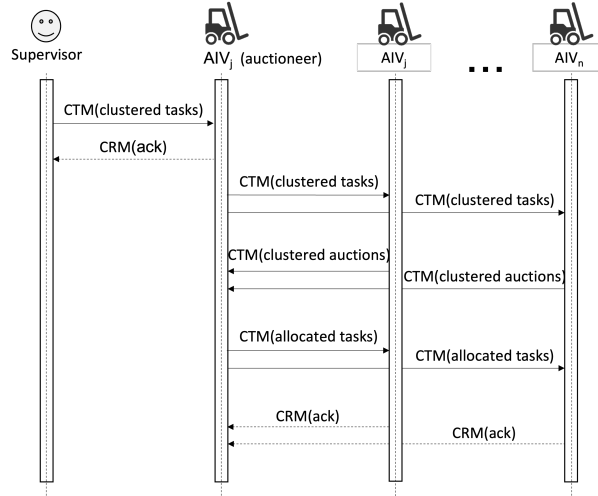


Figure 3.2 – CTM and CRM exchanged during TA

TA process. Then, we will provide the environment simulation proposed with the multi-agent simulation developed, before delving into the presentation of the different scenarios. Finally, we will present and discuss the results of these simulated scenarios obtained from these simulations.

3.3.1 Agent-based model simulation for TA scenarios

The ABMS architecture depicted in **Figure 3.3** is an augmentation of the model introduced in **chapter 2**. This advanced architecture is developed with the capacity of adaptation to model larger warehouses and to simulate problematic traffic situations involving a large number of AIVs. It will be used to test TA scenarios detailed in the **subsection 3.3.3**.

This chapter focuses on proposing collective strategies for TA. Specifically, we will define scenarios where AIV agents cooperate to optimize the performance of a set of missions, transmitted to them by a supervising agent. This one acts as an organizing service and these missions can be sent to the AIV in packets or continuous flow.

As in the previous chapter, each AIV agent has knowledge of the environment, i.e. the position of aisles, intersections, parking lots, storage points, battery replacement points and active elements of the infrastructure such as camera agents.

In this chapter, we make the strong assumption that active elements of the infrastructure, specifically cameras, participate in the cooperation. Indeed, they can particularly

contribute to the safety of AIV travel. This assumption of cooperative perception will be studied in greater depth in the **chapter 4**. Finally, to ensure this cooperation based on inter-agent communications in V2X mode, different types of standardized messages are used: CAM, DENM, CPM, MCM, and also our proposed standardized messages - CTM and CRM.

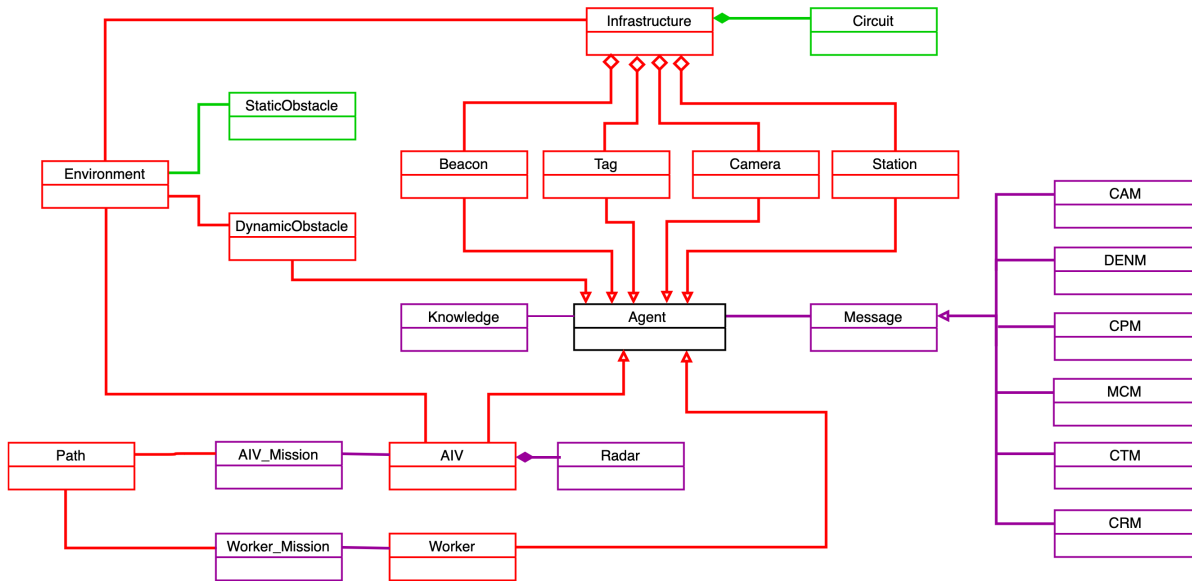


Figure 3.3 – Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple

3.3.2 Environment simulation

In this chapter, the environment chosen to illustrate our scenarios of problematic situations is the typical warehouse of [TBS18] adapted. This one was already presented in the **chapter 2**, in **section 2.3.2**. Certainly, this environment is small, but it allows us to detail very finely and from an educational point of view all the scenarios that we have defined, in particular the four scenarios studied in the **section 3.3.3**.

As the same work presented in **chapter 2**, five AIV agents are integrated into this environment corresponding to the five parking spaces available in this environment. One of the major interests of simulation is to be able to test the size of the vehicle fleet. Also, if the flow of tasks proposed to the AIVs becomes too great, leading to waiting times that are too long for the allocation of these tasks, the simulation conditions could be easily

adapted in the environment with the addition of new AIVs and of AIVs parking lots.

The costs in distance between the different nodes of the circuit are represented in the directed graph of **Figure 3.5**. The differences with scenarios simulation on obstacle avoidance presented in **section 2.3.2** are:

- Possibility of two-way traffic from node n_2 to n_{11} , whereas before it was one-way traffic.
- Setting up of cameras at node level n_{10} , n_{13} , n_{14} and n_{15} .

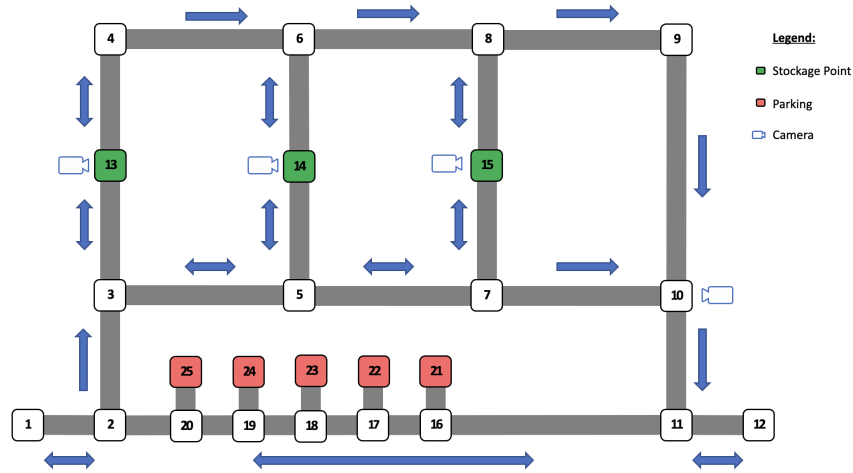


Figure 3.4 – Representation of the *IndustrialCircuit* for TA simulation scenarios

To enhance the visualization of task processes in these scenarios, we have upgraded the simulation interface from **Figure B.4**. The enhanced interface, shown in **Figure 3.6**, is designed to accommodate various traffic plans and adheres to the agent model illustrated in **Figure 2.2** depicted in **chapter 2 - subsection 2.2.2**. The different functionalities of this new Human-Machine Interface (HMI) TA application are described in **appendix B.4**.

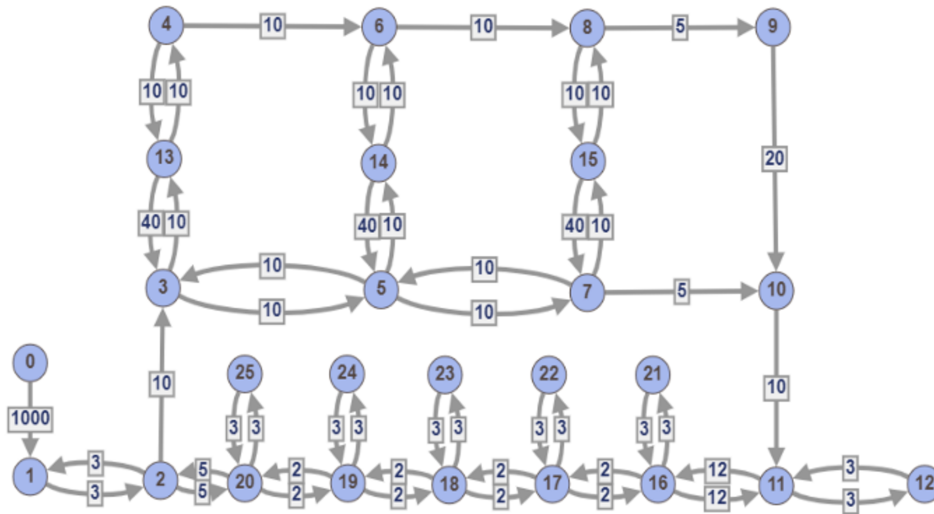


Figure 3.5 – Directed graph corresponding to the circuit

Warehouse - Wholesaler

Supervising VA					
VA Name	VA1	VA2	VA3	VA4	VA5
Available					
Pending Missions					
Number Tasks Validated	0	1	0	1	1
Status	circulates	circulates	stop	circulates	circulates
Mission Assigned	True	True	True	True	True
Mission On Going Part	1	2	1	2	2
Target Node	14	13	15	14	13
Target Validated	False	True	False	True	True
Current Segment	[5, 14]	[4, 6]	[20, 2]	[14, 6]	[6, 8]
Short Version Path Mission	[21, 21, 14, 25]	[22, 22, 13, 22]	[23, 23, 15, 21]	[24, 24, 14, 23]	[25, 25, 13, 24]
Number Mission Cycles	1	1	1	1	1

Supervising Tasks			
Task	Task Id	Validated Task	Status Task
[23, 15]	1	False	blocked
[25, 13]	2	True	terminate
[22, 13]	3	True	terminate
[24, 14]	4	True	terminate
[21, 14]	5	False	in progress
[15, 21]	6	False	attribute
[13, 24]	7	False	in progress
[13, 22]	8	False	in progress
[14, 23]	9	False	in progress
[14, 25]	10	False	attribute

Console
Simulation Size obstacle STOP
Scenario 0 1
Scenario 1 Generate obstacle Circuit model
Scenario 2
Scenario 3 Simulate failures
Scenario 4
Random Scenario

Supervising Camera

Camera Name	Camera1	Camera2	Camera3	Camera4
Status	working	working	working	working
Position	[285, 215]	[35, 115]	[135, 115]	[235, 115]
Detection	[]	[]	[]	[235, 115]

Figure 3.6 – Multi-agent simulation interface

3.3.3 Scenarios

In the previous chapter, we proposed AIV agent blocking scenarios that highlighted the need to increase cooperation/communication between agents if we wanted to effectively

132

manage these problems (**section 2.3**). We also discussed in the **section 2.2**, that the use of MCM, CAM and MCM made it possible to respond effectively to the problem of avoiding obstacles and collisions between AIV agents.

To verify the ability of AIV agents to carry out their missions while cooperatively managing the problems of obstacles or AIV agent breakdown, we will define a nominal case scenario and three scenarios:

- S_{c_1} : obstructed aisle;
- S_{c_2} : AIV breakdown;
- S_{c_3} : inaccessible stock point;

The description of these following scenarios includes five AIV agents, simulating five real AIVs of the same type, and illustrate the different types of V2X cooperation/communication that allow agents to anticipate problems and thus improve the performance of their missions collectively: V2I for the first scenario, V2V for the second, and Vehicle-To-Pedestrian (V2P) for the third. All information concerning the communication during these scenarios is resumed in the **Table 3.1**. In these scenarios, AIVs perform simple tasks:

1. load goods at a source storage point, then
2. drop them off at a destination storage point.

Scenario 1: obstructed aisle

The first scenario S_{c_1} is represented in **Figure 3.7**. It makes it possible to test the contribution of the cooperation between the AIV agents and the infrastructure agents for the performance of the tasks entrusted to the AIV agents. In this scenario, the camera agent placed to monitor the area around storage point n°14 detects an obstacle and sends a CPM message to the five AIV agents. As previously mentioned, we made the strong assumption that cameras can cooperate to evaluate our TA process.

The AIV agents, whose mission involves passing through the aisle obstructed by the obstacle, are able to re-plan their route in advance. Thus, this cooperation with the infrastructure makes it possible to avoid waiting for an AIV agent to detect the obstacle with its LiDAR and to warn the four other AIV agents when it passes near the obstacle (therefore saving time on detection). This scenario allows measuring the performance of a collective strategy including the infrastructure compared to a collective approach based only on V2V communications between AIV agents.

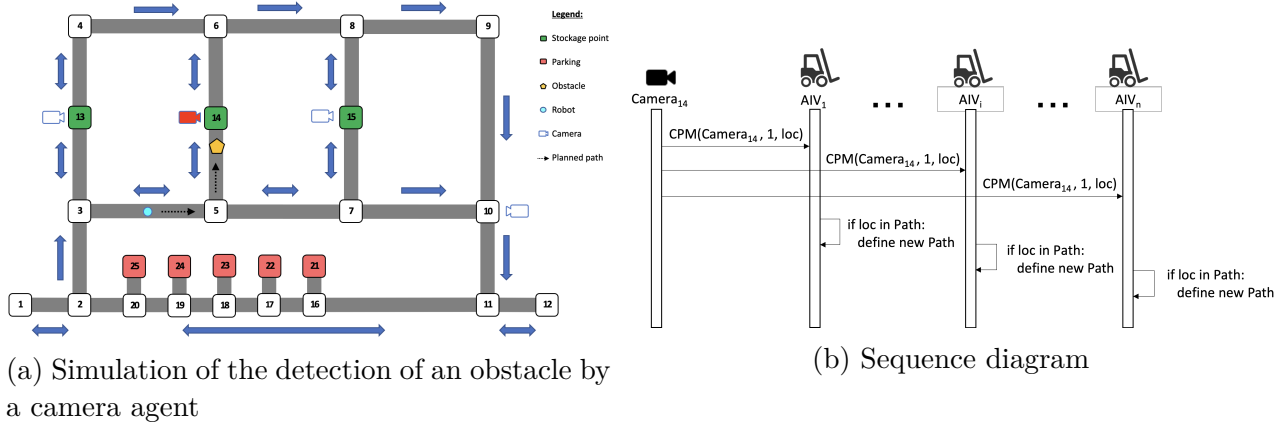


Figure 3.7 – Scenario S_{c1}

Scenario 2: AIV breakdown

The S_{c2} scenario corresponds to an inability for an AIV agent to complete its mission. This one can be blocked by obstacles or have a breakdown but without this preventing it from communicating (**Figure 3.8**). In this case, it is the dynamic TA mechanism presented in section 3, which is launched to reallocate the unfinished mission. The blocked AIV agent becomes an auctioneer. He transmits all the tasks he had to perform to the four other AIV agents using a CTM message. The AIV agents bid according to their situation and the tasks they are performing, which allows the AIV auctioneer agent to make its choice for the reallocation of the tasks he cannot complete.

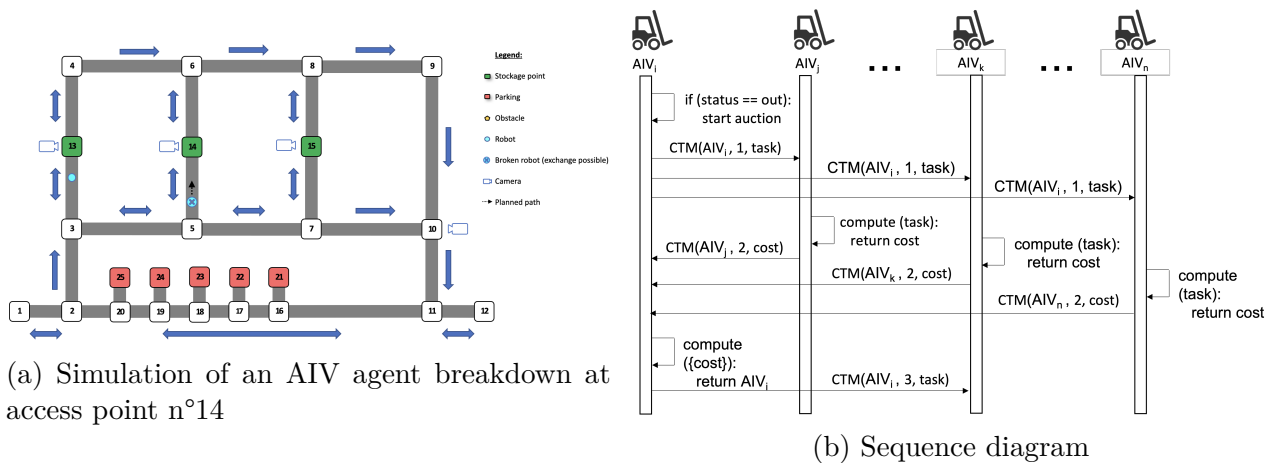


Figure 3.8 – Scenario S_{c2}

Scenario 3: Inaccessible stock point

Scenario S_{c_3} , depicted in **Figure 3.9a**, illustrates the ability of an AIV agent to handle the blocking problem when a task cannot be completed due to an event occurring on the warehouse circuit or in the defined environment to perform the task. It is further assumed that this blocking could not be detected by an infrastructure agent and therefore that its resolution was carried out by an AIV agent. For example, if a stock point designated as a target in the mission of an AIV_i agent is inaccessible (for instance, because of the presence of several obstacles), then the AIV_i agent must be able to inform the four other AIV agents that its mission cannot be carried out, using a CPM.

Subsequently, a human or an AIV_j agent having the ability to clear the obstacles in the aisle can intervene in response to the request made to it by the AIV_i agent, by sending a DENM (**Figure 3.9b**). The task that could not be performed before the human intervention is put back to auction as soon as the human has informed the AIV_i agent that the aisle is clear again, by sending a CPM. The AIV_i agent then temporarily becomes an auctioneer to manage the reallocation of the task. This prevents the AIV_i agent from waiting for human intervention to be able to continue its mission, and possibly complete another task.

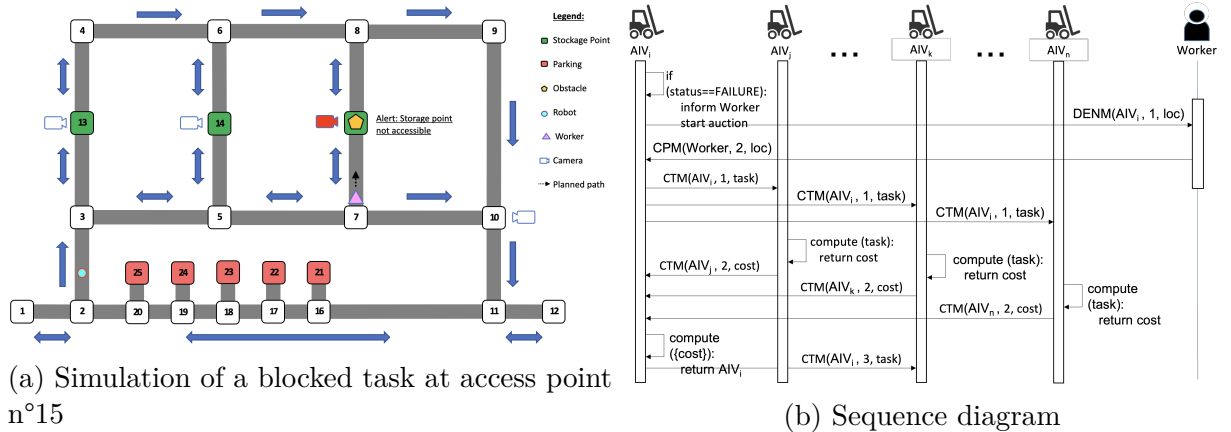


Figure 3.9 – Scenario S_{c_3}

3.3.4 Results

The three scenarios presented in the previous section were tested with the same dataset. The different types of V2X communications illustrated in the scenarios are summarised in the **Table 3.1**.

Scenario	Problem / Solution	Descriptive Figure / Result Table	Communication
S_{c_0} Nominal case	AIVs perform all tasks (mission, allocation, path planning) without encountering any problems.	Figure 3.4: Representation of the circuit Table 3.2: Test sets for the 3 scenarios	I2V: Supervisor gives tasks to auctioneer AIV V2V: CTM and CRM for the task allocation (auctions, reception and feedback)
S_{c_1} Obstructed aisle	If an obstacle obstructs an aisle, then an AIV that has to cross this aisle must quickly replan its path. Cooperation: AIVs and infrastructure	Figure 3.7: Simulation of the detection of an obstacle by a camera agent Table 3.3: Obstacle detected by an AIV Table 3.4: Obstacle detected by a camera	I2V: - Supervisor gives tasks to auctioneer AIV - Camera send a CPM 'Obstacle detected' V2V: CTM and CRM for the TA (auctions, reception and feedback)
S_{c_2} AIV breakdown	An inability for an AIV to complete its mission (obstacles or breakdown). Cooperation: AIVs (for sharing missions).	Figure 3.8: Simulation of an AIV breakdown at access point n°14 Breakdown of an AIV mission during part 1 (Table 3.5) or part 2 (Table 3.6)	I2V: Supervisor gives tasks to auctioneer AIV V2V: - CTM and CRM for the TA (auctions, reception and feedback) - CTM to share missions
S_{c_3} Inaccessible stock point	An inability for an AIV to complete its mission due to an event occurring in the defined environment to perform the task (stock point is inaccessible). Cooperation: AIVs and Worker (to remove an obstacle).	Figure 3.9: Simulation of the access point n°15 blocked Table 3.7: Stock point n°15 has become inaccessible Table 3.8: Stock point n°15 has become inaccessible, and then cleared by a worker.	I2V: - Supervisor gives tasks to auctioneer AIV - Camera send CPM: obstacle detected I2P: CPM for an obstacle at stock point V2V: - CTM and CRM for the TA (auctions, reception, and feedback) - CTM to share missions V2P: DENM for a blocking problem P2V: CPM indicating no more obstacle

 Table 3.1 – The various V2X communications in scenarios: S_{c_0} , S_{c_1} , S_{c_2} and S_{c_3}

The choice therefore fell on an allocation of tasks by packet, rather than continuously. A supervisor agent sends 10 tasks to an available AIV_i agent (when the AIV agents have no more missions to perform, they inform the supervisor agent). The AIV_i agent starts by clustering the tasks in missions and then offers them up for auction. The four scenarios were analysed with the following performance indicators:

- Tasks to be performed by each AIV agent: *tasksToBePerformed*.
- Tasks fully completed by each AIV agent: *tasksCompleted*.
- Ratio of the number of tasks to be performed to the number of tasks fully completed by each AIV agent: *ratio*.
- Total distance covered by each AIV agent: *distTravelled*.

Table 3.2 corresponds to the performance of all the tasks by the AIV agents, without them encountering any problem. This table includes the performance indicators listed above: the tasks requested and then allocated to each AIV agent, the tasks actually performed by the AIV agents, the ratio between the tasks allocated and performed, as well as the total distance covered by each AIV agent (distance in meters in this case).

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	<21, 14, 25>	21,14,25	2/2	124
AIV_2	<22,13,22>	22,13,22	2/2	116
AIV_3	<23,15,21>	23,15,21	2/2	112
AIV_4	<24,14,23>	24,14,23	2/2	114
AIV_5	<25,13,24>	25,13,24	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 22,13,22 23,15,21 24,14,23 25,13,24	10/10	580

Table 3.2 – Test sets for the scenarios

Results of Scenario 1

The results obtained during the execution of the scenario S_{c_1} corresponding to **Figure 3.7a** are given in **Table 3.2**. The obstacle was detected by the camera, which saves time on obstacle detection. Indeed, we tested this same scenario by disabling the camera agent. It was necessary to wait for the AIV_4 agent to arrive near the obstacle for it to be detected by its LiDAR. These results appear in **Table 3.3**. Thus, the total distance covered is 640 in the case where the camera agent detects the obstacle (**Table 3.4**), and 710 if the camera agent is deactivated (**Table 3.3**). This makes it possible to verify that cooperation with the infrastructure via camera agents can save time for the detection of obstacles, in particular by anticipating problems, and thus minimize distances for the performance of the missions of AIV agents.

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	<21,14,25>	<21,14,25>	2/2	164
AIV_2	<22,13,22>	<22,13,22>	2/2	116
AIV_3	<23,15,21>	<23,15,21>	2/2	112
AIV_4	<24,14,23>	<24,14,23>	2/2	204
AIV_5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 22,13,22 23,15,21 24,14,23 25,13,24	10/10	710

Table 3.3 – Obstacle obstructing an aisle, detected by the AIV_4 agent

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	21,14,25	[[21,14],[14,25]]	2/2	144
AIV_2	22,13,22	[[22,13],[13,22]]	2/2	116
AIV_3	23,15,21	[[23,15],[15,21]]	2/2	112
AIV_4	24,14,23	[[24,14],[14,23]]	2/2	154
AIV_5	25,13,24	[[25,13],[13,24]]	2/2	114
Global / Supervisor	[[23,15],[25,13]]	[[21,14],[14,25]]	10/10	640

Table 3.4 – Obstacle obstructing an aisle, detected by a camera agent

Results of Scenario 2

The scenario S_{c_2} was simulated in two cases:

- the delegation of a complete mission (two clustered tasks), and
- the delegation of the second part of a mission (only one task).

These delegations of a mission by an AIV agent can occur when the latter is unable to perform the mission in progress, following a breakdown or a blockage in an aisle for example. Thus, in **Tables 3.5 and 3.6**, it is possible to observe that the AIV_2 agent could not finalize its mission because its number of tasks performed is not equal to its number of tasks to be carried out. In **Table 3.6**, the AIV_2 agent was able to perform one task out of two of its missions.

It then started the task reallocation process, which resulted in the second uncompleted task being auctioned off. This task was won and performed by the AIV_1 agent. The latter therefore performed three tasks, whereas two tasks had initially been assigned to him. The second test for task delegation corresponds to **Table 3.6** where it is possible to see that the entire mission of the AIV_2 agent has been reallocated. In this case, it was the AIV_1 agent who took over the complete mission, while minimizing the overall distance covered.

Results of Scenario 3

The results of the simulated dataset with a problem accessing a stock point appear in **Table 3.7**. They correspond to the simulation of the S_{c_3} scenario with the blocking of stock point n°15 identified in **Figure 3.9**. We can notice that the AIV_3 agent who had two tasks related to the deposit or the retrieval of stock at stock point n°15 could not perform his tasks. Only eight of the ten tasks provided by the supervisor agent could be performed in this scenario. It is, therefore, necessary in this case, that an AIV agent or a human can come and unblock the situation (**Figure 3.9b**).

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	<21,14,25>	<21,14,25>	2/2	124
AIV_2	<22,13,22>		0/2	24
AIV_3	<23,15,21>	<23,15,21>	4/2	236
AIV_4	<24,14,23>	<24,14,23>	2/2	114
AIV_5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 23,15,21 22,13,22 24,14,23 25,13,24	10/10	612

Table 3.5 – Breakdown during part 1 of the AIV_2 mission

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	<21,14,25>	<21,14,25>	3/2	234
AIV_2	<22,13,22>	[22,13]	1/2	44
AIV_3	<23,15,21>	<23,15,21>	2/2	112
AIV_4	<24,14,23>	<24,14,23>	2/2	114
AIV_5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 [[13,22],[22,13]] 23,15,21 24,14,23 25,13,24	10/10	618

Table 3.6 – Breakdown during part 2 of the AIV_2 mission

The modified scenario, with human intervention and all tasks completed, is named Sc_3 . It is defined in **Figure 3.9b** and the simulation results are presented in **Table 3.8**. Furthermore, the supervisor agent can also be informed so that it does not request the performance of other tasks related to this storage point as long as it is not accessible.

3.3.5 Discussion on agent properties

We have employed the agent paradigm in our simulations, and this subsection will analyze the properties of the agents used across the different scenarios and explore the benefits they provide. Indeed, some properties can be associated with the concept of agent: situated, social, flexible, proactive, and robust [MG19]; but also, mobile, intelligent, rational, temporally continuous, coordinative, cooperative, competitive, rugged (able to deal with errors and incomplete data robustly) [Ode+00].

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	21,14,25	21,14,25	2/2	124
AIV_2	22,13,22	22,13,22	2/2	116
AIV_3	23,15,21		0/2	22
AIV_4	24,14,23	24,14,23	2/2	114
AIV_5	25,13,24	25,13,24	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 22,13,22 24, 14,23 25,13,24	8/10	490

Table 3.7 – Stock point n°15 has become inaccessible

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	21,14,25	21,14,25	2/2	124
AIV_2	22,13,22	22,13,22	2/2	116
AIV_3	23,15,21	23,15,21	2/2	112
AIV_4	24,14,23	24,14,23	2/2	114
AIV_5	25,13,24	25,13,24	2/2	114
Worker	1,15,1	1,15,1	2/2	116
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 22,13,22 23,15,21 24,14,23 25,13,24	10/10	580 AIVs + 116 Worker

Table 3.8 – The stock point n°15 has become inaccessible, then cleared by a worker (human operator)

The three scenarios presented in this section make it possible to verify the relevance of agent-based simulation. Indeed, all of the above properties are addressed during the realization of these scenarios (**Table 3.9**), where the AIVs agents will:

- carry out their missions in an environment where they will be located;
- communicate with each other, with the infrastructure and with workers, to establish collective intelligence;
- pursue a common objective of carrying out all tasks by cooperating with each other, with the infrastructure or with a worker;
- re-plan their paths and missions, or reallocate tasks, if necessary;
- listen to other AIVs and active elements of infrastructure, and continue to act even

- if they are blocked;
- coordinate themselves by using an auction mechanism for the allocation of tasks;
- collectively check possibly incorrect information and communicate with a worker to resolve any problems;
- act even when having incomplete data when receiving information without the AIVs being able to verify it themselves.

Scenarios	Agent properties												
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13
Sc_0	●	●	◐		◐	●	◐	●	●		●		
Sc_1	●		●		●	●	◐	●	●		●		◐
Sc_2	●	●	●	◐	◐		●	●	◐	●	●	●	◐
Sc_3	●	●	●	●	◐		●	●	◐	●	●	●	◐
Sc_4	●	●	◐	◐	●	●	●	●	●		●		●

Table 3.9 – The symbols means agent properties are captured ● or partially captured ◐, and the properties are: P1 – Situated; P2 – Social; P3 – Flexible; P4 – Proactive; P5 – Robust; P6 – Mobile; P7 – Intelligent; P8 – Rational; P9 - Temporally continuous; P10 – Coordinative; P11 – Cooperative; P12 – Competitive; P13 - Rugged

3.4 Possible impact on TA: trust on different messages

In the previous section, we verified the process of TA for AIVs, assuming that all exchanged messages were truthful. However, in the context of a warehouse for example, the presence of a faulty agent within the infrastructure, or an AIV transmitting unsolicited or incorrect messages due to sensor miscalibration or other issues, is a plausible scenario.

Therefore, in this section, we will address a specific problem related to message trust: *the issue of a faulty camera within the infrastructure*. We will explore a solution designed to detect this issue promptly, ensuring it does not disrupt warehouse operations for an extended period. By implementing this solution, we aim to maintain the integrity and efficiency of the warehouse environment, even in the presence of faulty equipment.

3.4.1 Scenario

Scenario Sc_4 presents a situation similar to Scenario Sc_3 described in the **section 3.3.3**, where an AIV agent encounters what seems to be an obstacle, as detected by a camera. However, in this instance, the camera’s assessment is faulty, and there is no actual obstruction present, as depicted in **Figure 3.10**.

Upon detecting the apparent obstacle, the camera notifies both the five AIV agents and the worker. However, upon closer inspection, it is revealed that the obstacle does not exist. Subsequently, the worker promptly sends two messages to the five AIV agents: a CPM message confirming the absence of an obstacle at the specified location, and a DENM message alerting about the failure of the camera agent.

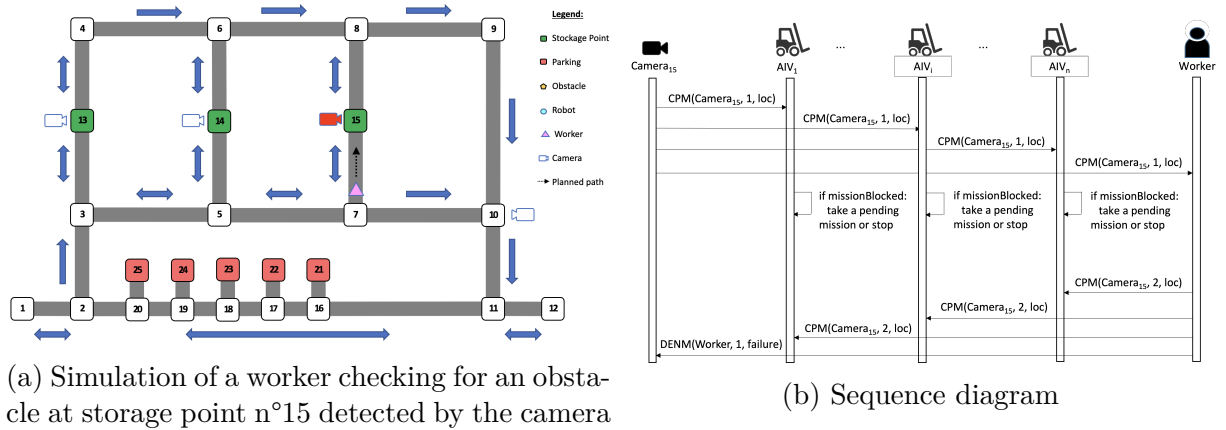


Figure 3.10 – Scenario Sc_4

Scenario	Problem / Solution	Descriptive Figure / Result Table	Communication
Sc_4 Camera failure	If a camera sends false information, this information has to be checked. Cooperation: AIV and Worker or between the AIVs (to verify information sent by the camera)	Figure 3.10 Simulation of a worker checking for an obstacle at stock point n°15 Table 3.11: A worker checks the presence of camera-detected obstacle	I2V: - Supervisor gives tasks to auctioneer AIV - Camera send CPM: obstacle detected I2P: CPM for an obstacle at stock point V2V: - CTM and CRM for the TA (auctions, reception, and feedback) - CTM to share missions V2P: DENM for a blocking problem P2V: - CPM indicating no more obstacle - DENM for the camera failure

Table 3.10 – The various V2X communications in the scenario Sc_4

3.4.2 Results

The results of the scenario S_{c_4} presented in **Figure 3.10** are revealed in **Table 3.11**. Results are the same as those in **Table 3.7**. However, the time differences compared to the scenario S_{c_3} occur if there are tasks continuous tasks. Because the AIVs will continue to pass through this area, and they will not have to wait for clearance by a worker or another robot related to an obstacle, as in scenario S_{c_3} . This scenario S_{c_4} demonstrates qualitative gains and robustness in processing. It highlights the importance of verifying information provided by the infrastructure because it can be faulty. This scenario underscores the importance of reliable infrastructure and effective communication channels among human operators, AIVs, and surveillance systems to address inaccuracies and ensure operational efficiency.

Agent	tasksToBePerformed	tasksCompleted	ratio	distTravelled
AIV_1	21,14,25	21,14,25	2/2	124
AIV_2	22,13,22	22,13,22	2/2	116
AIV_3	23,15,21		0/2	22
AIV_4	24,14,23	24,14,23	2/2	114
AIV_5	25,13,24	25,13,24	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	21,14,25 22,13,22 24, 14,23 25,13,24	8/10	490

Table 3.11 – A worker checks the presence of camera-detected obstacle

3.5 Conclusion

In the context of modern smart factories, mobile robots need to become increasingly autonomous to perform their tasks effectively. This autonomy allows them to optimize their operations based on various performance indicators, such as distance travelled, energy consumption, time taken to complete tasks, and system availability. Autonomy and decentralization are closely linked: an autonomous system makes decisions independently, while a decentralized system does not rely on a central authority for decision-making.

In the initial **section 3.2**, we proposed a novel framework for TA that facilitates better communication and collaboration among AIVs. By introducing new standard message

types specifically designed for collective task management, we addressed existing limitations in current communication protocols and bridged significant gaps in the TA process. These messages are pivotal for achieving "Level 3" on the collective autonomy scale (presented in **subsection 1.3.3**), as they enable collective decision-making for TA across the fleet. We propose a model that ensures tasks are allocated and executed according to some requirements (such as priority and time constraints), thereby enhancing the overall performance of the fleet. We also proposed a dynamic task (re-)allocation process model that effectively manages AIV activities within a decentralized framework.

In the subsequent **section 3.3**, we present an enhanced version of the agent model introduced in **chapter 2**, specifically developed to simulate the task (re-)allocation process. Subsequently, we introduced the upgraded multi-agent application, also discussed in the previous chapter, to simulate this process and evaluate its performance across various challenging traffic scenarios. The different scenarios are simulated with a fleet of homogeneous mobile robots in an industrial context. The different results obtained in simulation, demonstrated the practical effectiveness of the proposed TA strategies, showing how they adapt to real-world situations and optimize task distribution. The results highlighted the advantages of a collective approach, showcasing how AIVs can dynamically adjust their actions and reallocate tasks to address changing demands. The scenarios we tested not only advanced strong cooperation among AIVs but also facilitated interaction between AIVs and infrastructure agents, such as cameras. The V2X communication implemented to enable this cooperation is an essential element of our decentralized agent-based simulation approach. We have shown that it brings more flexibility and robustness in the management of problematic dynamic situations.

Finally, we addressed a critical challenge related to message trust and fault detection within the TA process. By focusing on the issue of a faulty camera and exploring solutions for prompt detection and mitigation, we aimed to ensure that the integrity of warehouse operations is maintained despite potential equipment malfunctions. The **section 3.4** highlighted the importance of robust fault tolerance mechanisms in preserving the efficiency and reliability of AIV fleets.

In summary, this chapter provides a comprehensive analysis of collective TA strategies, covering the development of a theoretical model, practical simulations, and fault management within the framework of dynamic TA strategies in V2X cooperation mode with infrastructure. By incorporating advanced V2X communication mechanisms, we have introduced a formal TA model that significantly improves the coordination and effectiveness

of AIVs fleets. The integration of robust communication, dynamic task management, and fault detection creates more resilient and adaptable autonomous systems, allowing them to perform effectively in complex and changing industrial environments, thus marking a substantial step towards "Level 3" on collective scale autonomy.

The different perspectives that emerge from this chapter are data fusion and shared memory of AIV agents. For example, how to merge data related to the detection of an obstacle by an AIV agent and by a camera agent at different times. We also aim to improve obstacle verification processes by having an AIV agent physically investigate the presence of an obstacle. Implementing shared memory would enable AIV agents to access global information, such as task delegation requests and to map the environment. To achieve this, we plan to suppress CRM and to choose to return to all AIV agents the mission assigned to them. These prospects for enhanced cooperation would make it possible to increase the autonomy and efficiency of AIVs. Additionally, another future work is to continue to develop the simulation platform to support fleets of heterogeneous robots, including those unable to perform all predefined tasks.

COOPERATIVE PERCEPTION STRATEGIES

4.1 Introduction

In the domain of AIVs, ensuring precise and comprehensive environmental perception is essential for safe and efficient operations. While autonomy can be enhanced through existing technical solutions—such as advanced sensors, data processing algorithms, and machine learning techniques—a more profound improvement can be achieved through a cooperative or shared vision among connected industrial systems.

Cooperative perception systems integrate data from multiple sensors and can be sent by V2X communications to achieve this goal, as presented in our previous chapters. These systems enable AIVs to detect and track various objects, including pedestrians and other vehicles, by combining information from diverse sources such as LiDAR, and infrastructure sensors (for instance cameras as in the **chapter 3**). By pooling data from different perspectives, a fleet of AIVs can construct a more accurate and comprehensive view of the environment, thereby enhancing collective autonomy.

The individual autonomy of an AIV is undoubtedly important, but its full potential is realized when it operates within a cooperative framework that enhances collective autonomy. In such a system, AIVs not only rely on their onboard sensors but also benefit from data shared by other vehicles and infrastructure, leading to improved situational awareness and decision-making capabilities.

This integrated approach relies on advanced techniques for matching sensor readings to specific objects (data association) and combining this data into a unified set (data fusion). Although our current work does not focus on these aspects, it's important to recognize that they play a crucial role in enhancing object detection accuracy and overall environmental awareness. This is particularly essential for navigating complex and dynamic industrial environments where safety and efficiency are paramount.

In this chapter, our focus shifts to acquiring road data using C-ITS, which can be provided by CAVs and infrastructure equipped with cameras. The objective is to obtain V2X communications corresponding to scenarios in the road or industrial field to improve the understanding of the surrounding environment for CAV.

By combining sensor data with V2X communications, cooperative systems can significantly enhance the safety and effectiveness of AIVs in dynamic environments. This is particularly relevant for reintegrating VRU, as discussed in the state-of-the-art review in **chapter 1, subsection 1.2.5**. Although we will not delve into the intricacies of data association and fusion in this work, our contribution will focus on exploring the cooperative context presented in **section 4.2**.

This entire section will set the context, before illustrating our process of acquiring essential data to generate V2X messages in the Carla simulator [Dos+17] in **section 4.3**. Moving on to **section 4.4**, we introduce an architecture designed to enhance cooperative perception by generating V2X messages, contributing to pedestrian safety, especially at crossroads. At the end of this section, we will summarise the proposed architecture and discuss potential directions for future research.

This chapter is related to the VTC-Fall conference paper [VTC24].

4.2 Cooperative perception context

In cooperative perception systems for AV, multiple sensors are deployed to detect various attributes of objects, such as position, velocity, acceleration, and type. These objects often include pedestrians, whose accurate detection and tracking are crucial for vehicle safety and navigation. As different sensors may capture data for the same pedestrian, the system receives numerous, potentially overlapping data points. In previous chapters, where we explored collective obstacle avoidance strategies and collective task allocation strategies, we made a key assumption: that active infrastructure elements, particularly cameras, are integral to cooperative operations. This assumption emphasizes the critical role of perception in enhancing the safety, navigation, and efficiency of a fleet of AIVs in performing their tasks, and it will be examined in greater depth in this chapter.

To manage the overlapping data points, data association is employed to match each observation to the correct pedestrian. Subsequently, data fusion techniques are used to combine these observations into a single, unified dataset. This process enhances the accuracy and reliability of the information, ensuring that the AV can effectively track and

respond to pedestrians. By leveraging data association and fusion, cooperative perception systems can improve their ability to reintegrate and accurately monitor pedestrians, thereby enhancing overall safety and efficiency in AV operations.

Firstly, we will introduce the input data required for data fusion. Subsequently, we will propose an analysis of a cooperative map architecture in a certain situation in **subsection 4.2.2**. Finally, in **subsection 4.2.3**, we will present and discuss a global architecture context for V2X data fusion.

4.2.1 Input data

To achieve effective data fusion, information is gathered from multiple sensors and transmitted using V2X communications. These vehicular communications involve sending messages such as CAM, CPM, and DENM. This enables a comprehensive understanding of the environment, incorporating information not detected by the ego vehicle's sensors.

In cooperative perception systems, infrastructures, pedestrians, and vehicles exchange information. This data must be integrated to create a cooperative perception map, allowing all road users to understand their surroundings and detect objects beyond their direct line of sight. This interconnected approach defines a cooperative system.

The collaboration between vehicles and infrastructures enhances the visibility of both connected objects (via CAM and CPM) and non-connected objects (via CPM). Without this cooperation, a vehicle would rely solely on its sensors, limiting its perception to directly detectable objects. In many scenarios, potential hazards may be obscured by other objects and remain undetected by the vehicle's sensors alone. For example, in the **Figure 4.1** the green AIV cannot see the pedestrian directly. However, it can be informed about the pedestrian's presence through communication with the orange AIV, or via the infrastructure with the camera.

Therefore, V2X communications augment and validate the data from the vehicle's onboard sensors, providing a more complete and accurate situational awareness. V2X data also facilitates predictive analysis to preemptively identify and mitigate potential road hazards. CAM and CPM are transmitted approximately every 0.1 seconds, allowing the system to process data, perform calculations, and draw timely conclusions.

Additionally, in the road field, roadside cameras detect and classify objects, transmitting this information to RSUs in the road field. RSUs can then convert this data into CPMs, sharing camera data through V2X communications. Moreover, the same process can be applied to data from LiDAR and RADAR sensors, further enriching the coop-

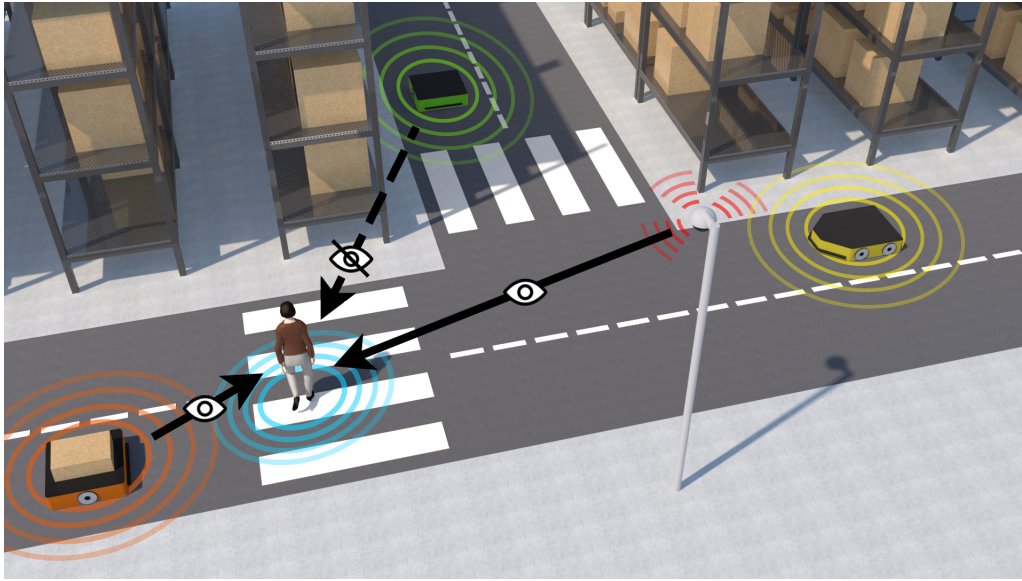


Figure 4.1 – Cooperative perception in a warehouse

erative perception map. This integration ensures that AVs benefit from a robust and comprehensive understanding of their environment, enhancing safety and efficiency on the road field.

We assume that information from infrastructure, such as cameras, can be effectively communicated in an Industry 4.0 context.

4.2.2 Cooperative map architecture analysis

To create a real-time map of each individual's position through V2X communication, the system can utilize CAM and CPM from various sources, including CAVs, pedestrians, and infrastructure.

As previously described for the industrial context, in **chapter 2, subsection 2.4.1**, these messages contain detailed information such as position, velocity, and direction, which can be fused to produce a more accurate and comprehensive situational picture. Data fusion improves accuracy, fills in missing data, and enhances overall environmental understanding. However, several challenges arise:

1. **Data Association:** recognizing which observations belong to which objects is crucial. Without this, merging data correctly is difficult, leading to potential calculation errors.
2. **Data Imprecision:** errors from faulty sensors or adverse weather conditions can

affect data accuracy. Identifying and correcting or discarding erroneous data is essential to avoid computation errors.

3. **Latency:** sensors may send data at different frequencies. For real-time systems, it's critical to merge data simultaneously despite these discrepancies.

Figure 4.2 depicted the situation of the **Figure 4.1**, where multiple sources send CPM or CAM related to the position estimation of the pedestrian. Input data are provided by a camera, the orange AIV, the green AIV and possibly the pedestrian. Indeed, the pedestrian may be equipped with a tracker sending CAM.

In this example, all stations provide nearly identical data, such as position, velocity, yaw, yaw angle, and direction, for the same object—the pedestrian. The accuracy of each message can vary depending on the sensor's precision and the reliability of the infrastructure and can be associated with an interval of confidence. This data can then be fused, resulting in a single CPM. The fused output will have improved accuracy compared to the input data, providing a more precise position for the pedestrian. The yellow AIV, for example, will receive accurate information about the pedestrian's position, which it cannot see directly, and integrate it into its localization map to avoid collisions.

In summary, utilizing V2X communication and fusing data from CAM and CPM significantly enhances the accuracy of position estimation for objects such as pedestrians. This improved situational awareness is essential for the safe and efficient operation of autonomous systems in dynamic environments.

4.2.3 Global architecture for V2X data fusion

Data association

One of the key aspects of data fusion using multi-sensor systems is recognizing which data points correspond to the same object. In this context, data association refers to grouping data from various sensors, such as RADAR, LiDAR, cameras, and V2X communications, that concern the same object.

Multiple sensors often provide data about the same object but assign different object IDs. Initially, it is not clear which data should be merged. Therefore, before performing data fusion to determine the positions of objects, data association is necessary to match the data from different sensors. Several machine learning algorithms can be employed to achieve data association. Regarding some association techniques, we find two primary approaches to data association:

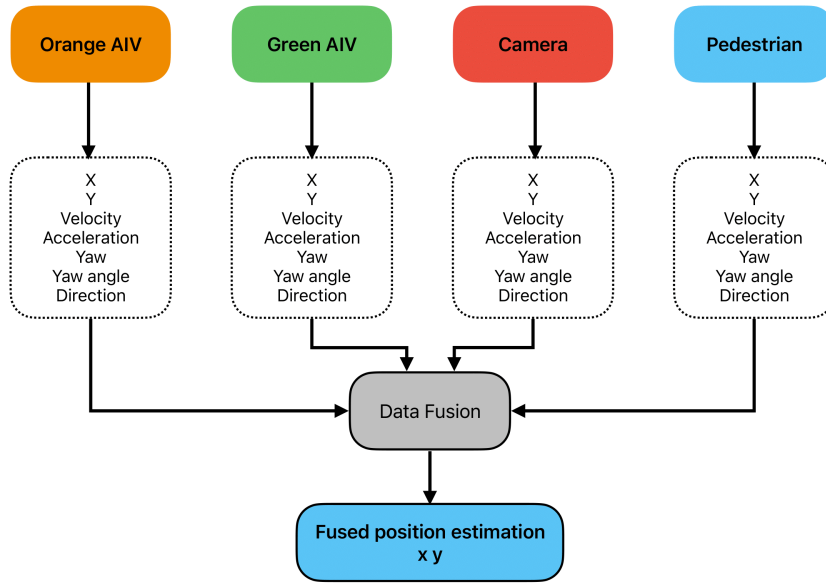


Figure 4.2 – Representation of data before and after fusion

- **Direct association without predefined clusters:** This method uses algorithms that do not require the number of clusters (objects) as input. These algorithms perform data association directly based on the data itself.
- **Cluster-based association with a predefined number of objects:** This method first involves estimating the number of objects present. This estimate is then used as the input for algorithms that require the number of clusters. In our context, the number of detected objects corresponds to the number of clusters.

By utilizing these approaches, data association can be effectively achieved, facilitating accurate data fusion from multiple sensors.

Our point of view is that in a cooperative perception system, the number of objects to detect cannot be known in advance, making it impractical to compute the number of objects solely based on observations from all sensors. Indeed, pedestrians, for example, are not always equipped with a tracker capable of sending CAM. This implies that clustering algorithms requiring the number of objects as an input cannot be used in this context. Additionally, CPM contain information about multiple objects, and there are multiple CAV. This means that the association process must handle many objects simultaneously.

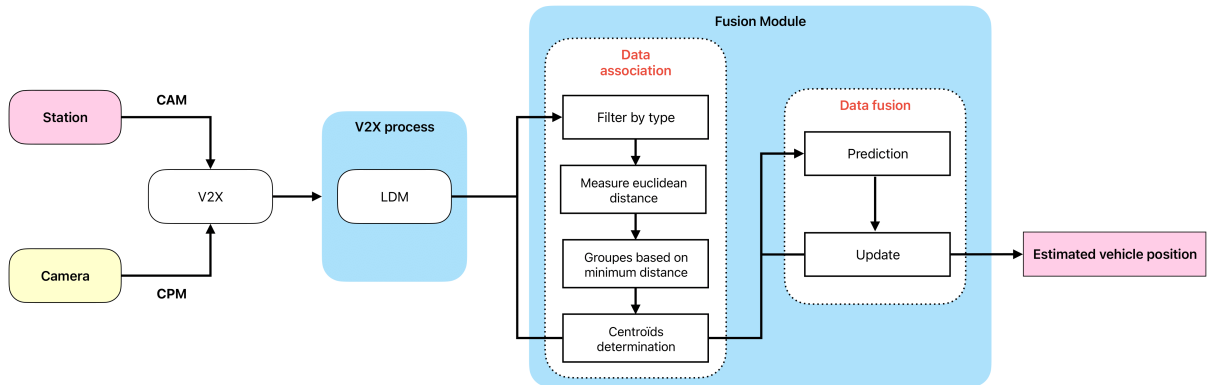


Figure 4.3 – Global architecture for data fusion

Data fusion

After identifying which observations belong to each object, the next step in the fusion module proposed in **Figure 4.3** is data fusion. Data fusion involves combining data from multiple sensors to enhance accuracy and reliability. By integrating information from various sources, such as GPS, cameras, RADAR, LiDAR, and V2X communication, the system can achieve more precise vehicle position estimates.

Currently, many sensors provide vehicle position estimates with uncertainties ranging from meters to centimeters. The goal is to merge data from all available sensors to improve the accuracy of these estimates. This multi-sensor fusion allows connected objects, such as vehicles, infrastructure, and pedestrians, to determine the exact location of nearby objects, whether they are connected or not.

There are numerous methods for performing data fusion, each with its strengths and weaknesses. Therefore, choosing the most appropriate method is crucial for optimizing the system's performance. However, as previously mentioned this thesis does not delve into association and data fusion. Nonetheless, understanding the significance of obtaining accurate V2X data is essential for enhancing CAVs' awareness of their surroundings. Indeed, accurate V2X data allows vehicles to gather information about nearby vehicles, pedestrians, road conditions, and other relevant factors in real-time. This awareness is crucial because it enables CAVs to make informed decisions, such as adjusting speed or anticipating potential hazards. Therefore, ensuring the accuracy and reliability of V2X data is essential for enhancing the overall safety, efficiency, and effectiveness of CAVs' operations and decision-making processes.

4.3 Architecture proposed in Carla Simulator

In the **chapter 1, subsection 1.4.5**, we present different realistic simulators to train autonomous road vehicles, and particularly Carla Simulator [Dos+17] and AWSIM¹. As outlined in the introduction, our objective is to gather road data through C-ITS, encompassing information from vehicles, infrastructures, cameras, and other sensors. Analyzing video streams from vehicles and infrastructure will generate valuable data for AI, aiding in the reintegration of VRU after information fusion.

This section aims to propose a solution for creating a realistic simulation incorporating V2X messages exchanged between vehicles and RSUs. However, Carla Simulator or AWSIM lack the richness of C-ITS services due to their inability to incorporate V2X communication. To address this limitation, we will utilize Carla Simulator in conjunction with OpenCDA [Xu+21a; Xu+23] which facilitates the acquisition of different information for the content of V2X messages.

Therefore, the objective is to generate ETSI messages in Carla Simulator. Subsequently, we will define the simulation scenario that will guide the rest of this chapter. Following this, we will present our approach to generate CAM and CPM using Carla Simulator, supplemented with result examples.

4.3.1 Scenario considered

Road vehicles must navigate through dense and diverse traffic, including VRUs, and their visibility is often constrained. This limitation can stem from factors such as the perception field, influenced by sensor type and prevailing weather conditions. Additionally, the increased distance to obstacles may reduce perception accuracy, and blind spots created by structures like vehicles or buildings can further impede the comprehensiveness of their perception.

Cooperative perception enables CAVs to achieve a comprehensive understanding of their surroundings, effectively eliminating blind spots in mixed traffic scenarios. In *Scenario 1* depicted in **Figure 4.4a**, a situation is illustrated where a pedestrian is crossing an intersection while a vehicle approaches from the opposite direction. However, the vehicle's view of the pedestrian is partially obstructed by a building at the corner, making it difficult for the vehicle to detect the pedestrian in time. In *Scenario 2* illustrated in **Figure 4.4b**, another vehicle obstructs the view between the approaching car and a bus

1. AWSIM: TIER IV inc. - <https://tier4.github.io/AWSIM/>

parked near the intersection. The bus creates a blind spot, making it challenging for the approaching car to notice a pedestrian or another vehicle that may suddenly appear from behind the bus.

In both cases, the car cannot directly observe the pedestrian, and vice versa. However, when cooperative perception is activated in collaboration with the infrastructure, the car gains the ability to "see" the pedestrian, enabling it to slow down or come to an urgent stop, especially if the pedestrian is crossing without paying attention to the road. The activation of cooperative perception is crucial for reducing the occurrence of accidents. We will utilize this application scenario, illustrating the utilization of cooperative perception, to aid in the seamless reintegration of VRUs into the broader traffic context.

4.3.2 CAM: Self-localization of vehicle

As seen before, cooperative perception allows us to incorporate VRU back into C-ITS. In practical terms, vehicles depend on Global Navigation Satellite System (GNSS) for their localization. The accuracy of GNSS may vary based on several factors. These include the type of equipment used, the GNSS constellations accessed, any corrections applied, and most importantly, the environment (whether it's urban or rural).

In our study, we used data from the Carla Simulator to generate CAMs. Specifically, we used the GNSS/Inertial Measurement Unit (IMU) fusion algorithm for localization, which is implemented in OpenCDA [Xu+21a; Xu+23]. An example of a CAM sent by the CAV shown in the *Scenario 2* in **Figure 4.4b** is presented in **Figure 4.5a**.

The example CAM message demonstrates the detailed structure of how localization and vehicle status information are encoded. Key fields such as *latitude_value*, *longitude_value*, and *heading_value* provide detailed information about the vehicle's exact location and orientation. These values are accompanied by corresponding confidence metrics (*latitude_confidence*, *longitude_confidence*, *heading_confidence*), which reflect the level of uncertainty in the GNSS-based localization process.

This uncertainty can arise from various factors, including environmental conditions and the effectiveness of sensor fusion. Understanding this data structure is essential for analyzing how location errors may propagate through the V2X communication system and impact overall reliability. We could apply an error model to mimic the typical location inaccuracies of GNSS systems. The content of V2X messages can be randomized to achieve varying levels of precision. For instance, a robot's localization system uses odometry instead of GNSS, which can result in less accurate data.



(a) Scenario 1



(b) Scenario 2

Figure 4.4 – Problematic scenarios: view of the camera infrastructure © IEEE 2024

4.3.3 Building CPM

Creating CPM involves identifying, localizing, and tracking objects within the Carla simulation to determine their paths. This allows the CPM to provide useful information to nearby vehicles. To generate the necessary data for CPM, we need to use software that can detect and track objects, analyzing the video stream produced by the roadside cameras.

This tracker ensures that the same IDs are used in each CPM for the same object, which is crucial for the subsequent steps of data association and fusion in V2X communication. As part of our contribution to the development of OpenCDA, we proposed a combination of the Deep SORT algorithm with the Yolo detection framework. Furthermore, we upgraded the existing Yolo V5 algorithm to Yolo V8 for better accuracy.

An example of a CPM sent by the RSU which provides a view of the intersection in **Figure 4.4** is presented in **Figure 4.5**.

The CPM example is particularly illustrative of the process of identifying and tracking multiple objects, such as vehicles and pedestrians. Each *perceived_object_container* entry represents a distinct object, with fields like *x_distance*, *y_distance*, and *z_distance* detailing the spatial position relative to the RSU. Additionally, the confidence values linked to each object's classification, indicate the algorithm's certainty in identifying the object's type (e.g., vehicle or pedestrian respectively type 5 or type 2). The consistency of object IDs (*object_id*) across CPM is essential for maintaining accurate tracking over time, which is critical for effective data association and fusion in V2X communications.

```

/CAV_138/CAM
custom_interfaces/msg/CAM @ 1702745521.131000000 sec
header {} message_id: 2, station_id: 138
message_id 2
protocol_version 2
station_id 138
cam_data {} 2 keys
generation_delta_time 8.97682323306799
cam_parameters {} 2 keys
basic_container {} 5 keys
station_type 5
latitude {} 2 keys
latitude_value -0.00013329209612322757
latitude_confidence 0
longitude {} 2 keys
longitude_value 0.0003122553196775132
longitude_confidence 0
altitude {} 2 keys
altitude_value 0.03244909213390201
altitude_confidence 0
position_confidence_ellipse {} 3 keys
high_frequency_container {} 1 key
current_basic_vehicle_container_high_frequency {} 9 keys
vehicle_length {} 2 keys
vehicle_width {} 2 keys
yaw_rate {} 2 keys
heading {} 2 keys
heading_value -88.24530029296875
heading_confidence 0
curvature_calculation_mode 0
longitudinal_acceleration {} 2 keys
drive_direction 0
speed {} 2 keys
value 0.06329817324876785
confidence 0
curvature {} 2 keys
    
```

(a) CAM sent by the CAV

```

/RSU_1000/CPM
custom_interfaces/msg/CPM @ 1704906063.605000000 sec
header {} message_id: 14, station_id: 1000
cpm_data {} 2 keys
generation_delta_time 8.858778800815344
cpm_parameters {} 4 keys
management_container {} 2 keys
sensor_information_container {} sensor_id: 0
perceived_object_container [] 3 items
0 {} object_id: 3, sensor_id: 0
object_id 3
sensor_id 0
time_of_measurement 0
x_distance {} 2 keys
y_distance {} 2 keys
z_distance {} 2 keys
x_speed {} 2 keys
y_speed {} 2 keys
z_speed {} 2 keys
yaw_angle {} 2 keys
classification [] 1 item
0 {} 2 keys
confidence 0.7738106846809387
class_object {} 2 keys
type 5
confidence 0.7738106846809387
1 {} object_id: 12, sensor_id: 0
object_id 12
sensor_id 0
time_of_measurement 0
x_distance {} 2 keys
y_distance {} 2 keys
z_distance {} 2 keys
x_speed {} 2 keys
y_speed {} 2 keys
z_speed {} 2 keys
yaw_angle {} 2 keys
classification [] 1 item
0 {} 2 keys
confidence 0.4999338388442993
class_object {} 2 keys
type 5
confidence 0.4999338388442993
2 {} object_id: 9, sensor_id: 0
object_id 9
sensor_id 0
time_of_measurement 0
x_distance {} 2 keys
y_distance {} 2 keys
z_distance {} 2 keys
x_speed {} 2 keys
y_speed {} 2 keys
z_speed {} 2 keys
yaw_angle {} 2 keys
classification [] 1 item
0 {} 2 keys
confidence 0.4922557473182678
class_object {} 2 keys
type 2
confidence 0.4922557473182678
number_of_perceived_objects 3
    
```

(b) CPM sent by the RSU

Figure 4.5 – V2X messages in ROS2 topics generated from the scenario in **Figure 4.4** © IEEE 2024

4.4 Evaluation of the global architecture

4.4.1 Implementation of the architecture

In the system architecture described in **Figure 4.6**, the process of generating and broadcasting V2X messages in the environment simulated on Carla with the collaboration of the OpenCDA framework is presented. We chose to work with different components running in Docker containers to simplify deployment.

General description

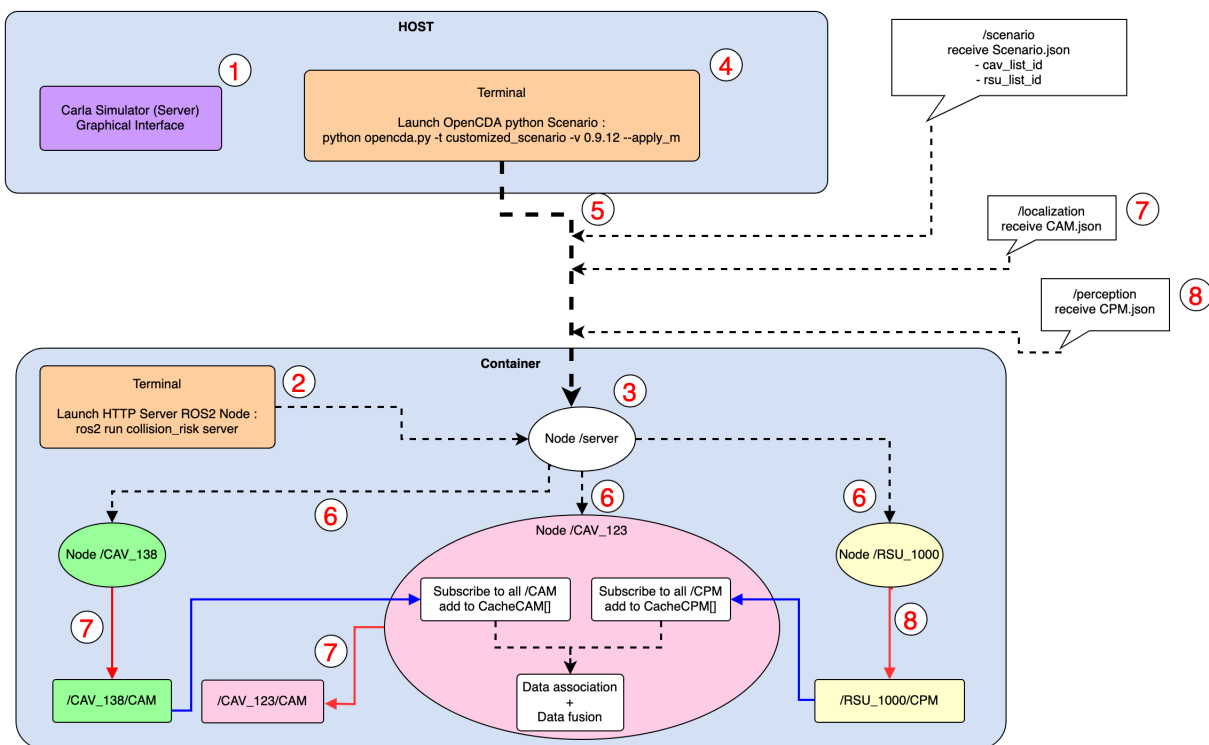


Figure 4.6 – Global architecture: Carla-OpenCDA-ROS2 © IEEE 2024

The workflow begins with the initiation of the Carla Simulator, configured with a pre-defined scenario featuring CAVs and RSUs, using the module ScenarioManager of OpenCDA. The ROS2 node server, embedded within the Docker container, acts as the intermediary by interpreting JSON-formatted messages from three different endpoints: `"/scenario"`, `"/localization"` and `"/perception"`.

All agents present in the scene are communicated with their unique identifier to an

HTTP server with the `/scenario` endpoint. Messages received in `/scenario` enable to dynamically launch the required nodes and topics, adapting the simulation environment in real-time.

This HTTP server, residing within a Docker container, receives and processes the information, extracting unique identifiers for both CAVs and RSUs. Subsequently, the ROS2 server dynamically generates nodes corresponding to each agent in the scenario and establishes the requisite communication topics. Notably, the topics are configured for CAM and CPM. Cameras exclusively publish to the CPM topic, while CAVs publish messages in both CAM and CPM, each linked to their unique identifier.

To conclude, this is the detailed process of the architecture presented in the **Figure 4.6**.

1. Launch of Carla Simulator: the graphical interface.
2. Launch of the HTTP Server ROS2 node inside the docker container.
3. Set up of the Node Server.
4. Launch of the scenario using the module ScenarioManager of OpenCDA.
5. Reception of the "Scenario.json" information via the endpoint `/scenario`.
6. Creation of every ROS2 nodes corresponding to all actors (CAVs and RSU) present in the scenario. Creation of every CAM ROS2 topics for each actor sending CAMs. Creation of every CPM ROS2 topics for each actor sending CPMs.
7. Reception of "CAM.json" via the endpoint `/localization` and diffuse the content within the relevant CAM topic of the agent concerned defined by the ID.
8. Reception of "CPM.json" via the endpoint `/perception` and diffuse the content within the relevant CPM topic of the agent concerned defined by the ID.

Description detailed with the specific scenario

The **Figure 4.7** illustrates the interactions between two CAVs, a RSU, and a pedestrian in the scenario shown in **Figure 4.4b**.

This sequence diagram details the communication exchange between the two CAVs (`CAV_123` and `CAV_138`) and the RSU (`RSU_1000`), which occurs in the presence of a pedestrian. The object detection process, which identifies pedestrians and vehicles using Yolo v8 and DeepSort algorithms, is communicated through CAM and CPM messages.

Furthermore, **Figure 4.8** illustrates the nodes and topics involved. In this scenario, the CAV is identified by ID 345, while the RSU, specifically the infrastructure camera, is identified by ID 1000.

This diagram shows the ROS2 communication flow, where CAM and CPM messages are exchanged between different nodes (*CAV_123*, *CAV_138*, and *RSU_1000*). It highlights the communication pathways and message exchanges within a V2X environment.

In the Carla simulator, CAV IDs directly correspond to those assigned within the simulator environment. However, for RSUs, IDs are defined differently in OpenCDA's YAML files that specify the scenario. To bridge this gap in the ROS2 framework, we have adopted a mapping convention where negative IDs in OpenCDA's YAML files are translated into positive IDs for ROS2:

- An ID of -1 in OpenCDA corresponds to an ID of 1000 in ROS2.
- An ID of -2 in OpenCDA corresponds to an ID of 1001 in ROS2.
- And so forth.

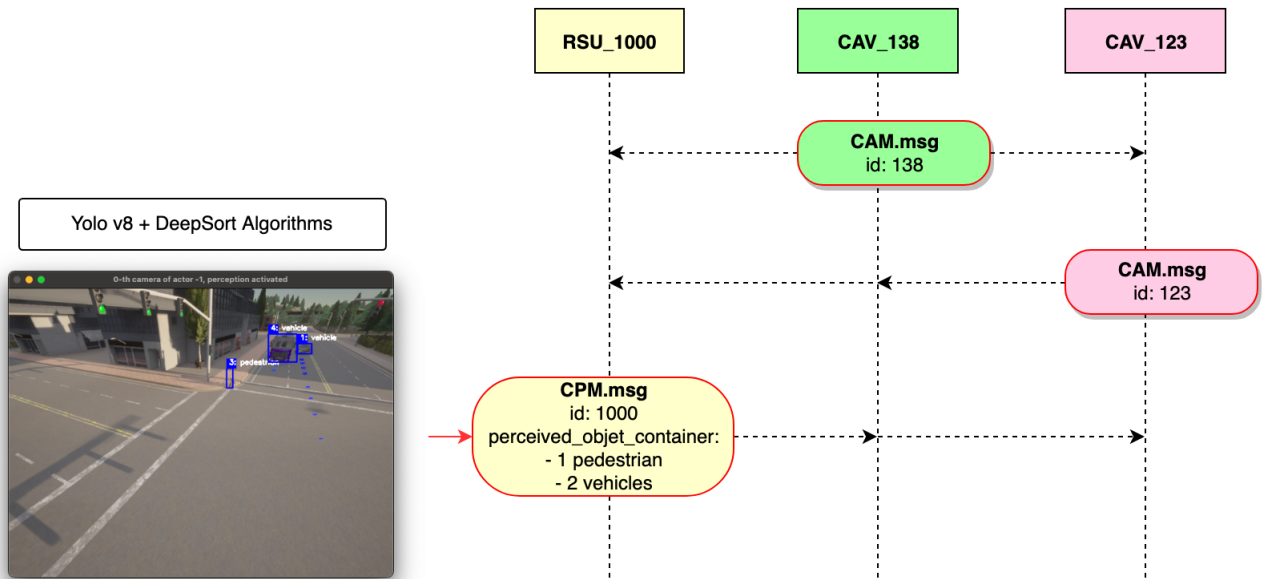


Figure 4.7 – Sequence diagram of the communication and object detection process in the scenario depicted in **Figure 4.4b** involving two CAVs and one RSU © IEEE 2024

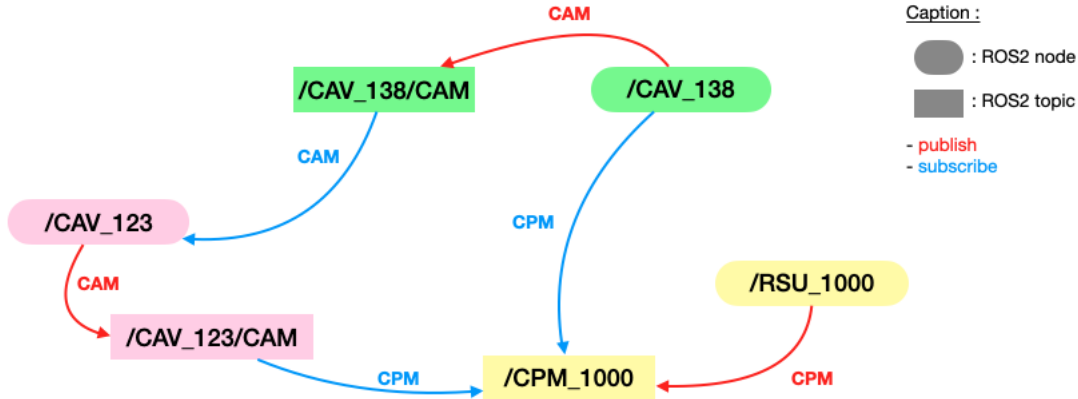


Figure 4.8 – ROS2 communication flow of CAM and CPM in the scenario depicted in Figure 4.4b © IEEE 2024

This communication infrastructure ensures precise connectivity for each agent. As the simulated scenario unfolds, OpenCDA enables the creation of V2X messages, specifically CAM and CPM, derived from the simulated data within Carla. These messages are then dispatched to designated endpoints, "/localization" for CAM and "/perception" for CPM messages.

Afterwards, individual agent nodes publish the received messages in the corresponding topic, ensuring that the information is disseminated accurately to the relevant entities based on their unique identifiers. This comprehensive framework is managed by ROS2 server within the Docker container, facilitating the seamless generation, transmission, and reception of V2X messages. This setup effectively simulates a realistic connected vehicle environment for advanced research and development purposes.

In order to facilitate community utilization, we are releasing the code of our project, enabling V2X messages generation tool for Carla Simulator².

4.4.2 Discussion

Our design empowers the utilization of the Carla Simulator for generating V2X datasets containing CAM and CPM produced by both CAVs and RSUs. The adaptability of our architecture permits users to implement their preferred algorithms to enhance message content, particularly for CPM. While our current approach involves using the latest YOLO version for object classification and tracking in the scene, alternative algorithms

2. GitHub page: <https://gitlab.inria.fr/jgrosset/carla-v2x-release>. A simulation video of the Scenario 1 presented in Figure 4.4a is also available at: <https://youtu.be/o081bVM9pB8>.

can be seamlessly integrated. Presently, we depend on the Carla server to retrieve GNSS data for each identified vehicle, facilitating the completion of CPM with precise orientation, speed, and position values. This arrangement enables the testing of algorithms for predicting and calculating the position of recognized objects from the video stream. In this iteration, we intentionally provided accurate data in CAM and CPM to assess V2X data fusion algorithms. The architecture’s flexibility allows us to concentrate on association and fusion mechanisms. However, to evaluate network operation, we would need to use a network simulator during message generation or include a message loss model. This would allow for the randomization of data availability to simulate more realistic scenarios. Our ultimate objective is to advance V2X data fusion in intricate traffic scenarios, involving multiple CAVs, vehicles, RSUs, and VRUs.

Finally, our system design offers the flexibility to investigate a variety of scenarios. For example, in the **subsection 4.3.1**, we examined a situation where the RSU, serving as an infrastructure camera, sends a CPM to improve the perception of a vehicle obstructed by a building in *Scenario 1*. Alternatively, in *Scenario 2*, we could analyze the video stream from a truck’s camera, which detects the pedestrian and transmits a CPM. In this case, the following car would receive information about a pedestrian crossing in front of the truck, even if the pedestrian is not directly visible due to the truck. As a result, such a service is not dependent on the availability of infrastructure. This flexibility allows us to evaluate different situations, enabling the exploration and study of various services or the same service with different deployment configurations.

4.5 Conclusion

In **chapter 3**, we showed that enhancing the perception of AIVs by adding cooperative elements led to better performance. This improvement was particularly evident in tasks such as obstacle detection and TA, where the shared information and coordination between agents led to more accurate and efficient decision-making. By leveraging cooperative elements, AIVs could better anticipate and respond to dynamic changes in the environment, thereby optimizing task execution. This chapter extends that discussion by focusing on how to effectively implement these cooperative perception strategies. Rather than re-proving the concept, we now aim to illustrate the practical application of these methods to achieve improved outcomes.

Despite advancements in CAVs, there remains a challenge by integrating into the

C-ITS framework due to the lack of C-ITS equipment. This deficiency impedes their interaction with CAVs. We underscore the significance of V2X communication in enhancing road safety with VRUs by facilitating information exchange between CAVs and the infrastructure. This communication is pivotal for reintegrating VRUs into the environmental awareness of CAVs. Indeed, generating V2X datasets can help the community to work on the refinement of algorithms for ADAS for example. To determine the essentials for generating accurate V2X information, we establish a cooperative perception context in the **section 4.2** with a vision of the global architecture for V2X data fusion.

Currently, the Carla simulator, widely used for AV training, currently lacks comprehensive V2X communication capabilities. In response, we proposed a tool that is useful to the ITS community for working with V2X data in the Carla environment [Dos+17]. Indeed, we have developed an architecture using Carla simulator, OpenCDA, and ROS2 to generate V2X messages based on ETSI standards. This architecture enables the creation of V2X messages sent by CAVs and RSUs in a C-ITS environment while performing realistic Carla simulations. It also simplifies the creation of V2X datasets. An example of its utility to the ITS community is testing V2X data fusion algorithms for ADAS or AVs.

The cooperative perception enabled by the sharing of information between CAVs and RSUs means that VRUs can be reintegrated into the C-ITS system. The cooperative messages will provide information on blind spots, which can be used by ADAS, in particular to avoid accidents. To illustrate and assess our proposed architecture, we present a scenario involving a pedestrian concealed in a blind spot for a connected vehicle in the **subsection 4.3.1**.

Industry may be interested in our tool since ROS2 is extensively used for robotic simulations. In the future, the use of cooperative mobile robots could occasionally be part of road environments, for example, in industrial activity parks. To manage cooperation and heterogeneity in such a fleet of mobile robots, the use of C-ITS ETSI messages brings many benefits [ICAE24]. It will allow cooperation between robots and infrastructure on the model of cooperative ITS and reduce reliance on central servers. To further evaluate this possibility, we investigated how to integrate industrial vehicles into Carla simulator. Thus, with the ROS2 environment, we will be able to emulate simulated robots in the Carla environment, and for example work on co-simulation [DF23].

In our future work, we are seeking to feed AI with relevant information. To achieve this goal, we plan to implement a data clustering approach that assigns confidence levels to each sensor’s input, such as those from cameras.

Eventually, we aim to associate a level of confidence directly with the data to make it dynamic. We think it should help to deploy a multi-stakeholder system allowing more adaptive and responsive systems. Additionally, Carla simulator can help us reduce the time required to deploy scenarios with active infrastructure. For instance, we can evaluate various camera placements and their impact on the assessment of the scene, and therefore on the quality of decisions made by robots. Finally, by utilizing a network simulator or a message loss model, we can observe how data fusion responds to changes in data freshness.

COLLECTIVE ENERGY MANAGEMENT STRATEGIES

5.1 Introduction

Energy management is a crucial determinant of the operational success of AIVs, as these systems are predominantly battery-powered. The operational efficiency, reliability, and longevity of AIVs are intrinsically linked to the optimization of battery life and recharging processes. Inefficient energy management can result in frequent operational interruptions, increased downtime, and elevated operational costs, thereby negating the potential benefits of automation. As the deployment of AIVs continues to expand, the urgency for developing robust energy management strategies becomes increasingly apparent.

This chapter concentrates on the development and integration of energy-efficient strategies designed to optimize the overall performance of AIV fleets. Given the non-uniform distribution of AIV missions—characterized by periods of high activity followed by intervals of reduced demand—it is imperative to align energy consumption with the actual workload and operational availability of the vehicles. In heterogeneous fleets, where each AIV may have distinct energy needs and capabilities, a collective and decentralized approach to energy management becomes essential. By allowing each AIV to autonomously monitor its recharging needs and collaborate with other vehicles, the fleet can more effectively manage energy resources and reduce the likelihood of operational disruptions.

To mitigate the risk of simultaneous recharging requests, which could overwhelm available resources, it is essential that AIVs cooperate. This can be achieved through inter-vehicle communication or via a centralized infrastructure. Although automatic recharging systems can address the frequency of recharges, they necessitate additional spatial and energy resources. Even a marginal reduction in energy consumption, such as 2 to 3%,

can have substantial implications in energy-intensive environments like warehouses and airports.

The rapid advancement in automated systems has revolutionized various industries, including the baggage handling sector. For the successful integration of AIV fleets in future industrial applications, it is necessary to optimize the number and distribution of recharging points, taking into account the potential for vehicle communication to preempt critical recharging scenarios. In heterogeneous fleets, this optimization must account for the varying energy demands and operational roles of different AIVs. By coordinating energy management collectively, AIVs can reduce the incidence of urgent recharging events, thereby maintaining continuous operations and minimizing downtime.

This chapter delves into the collective energy management strategies for AIVs, focusing on the integration of energy efficiency and TA to optimize overall system performance.

In **section 5.2**, we propose a fuzzy decision model for battery recharging. We use fuzzy logic to manage the uncertainties inherent in the recharging process.

Then, in **section 5.3**, the focus shifts to the simulation application of the fuzzy agent-based simulation for managing battery recharging. Through a case study, the autonomous management of battery recharging using fuzzy logic is examined, showcasing the real-world benefits of this approach. We conduct a comparative analysis between threshold and fuzzy logic models. Then, we present the results of three heuristics simulated in three scenarios, incorporating more realistic constraints of an airport, such as the flow of baggage arrivals.

To achieve a more realistic simulation framework, we further refine the AIV energy model. Therefore, in **appendix D**, we propose a methodology for energy management with a realistic point of view. We also describe in detail an algorithm for the energy consumed by each strategy and a fuzzy logic model to select the strategy adapted for each AIV.

Finally, in **section 5.4**, we address the problem of recharging batteries in conjunction with TA. This section integrates the concepts of TA and energy management, highlighting the interdependence of these two aspects.

This chapter is based on work published in [ASP24] and [AISYS24], as well as research currently under review [ALGO24], which has not yet been published.

5.2 Fuzzy decision model for battery recharging

On one hand, we present a formal definition for the different elements of a fuzzy agent model, as previously mentioned in **chapter 1 - section 1.4.4** [Fou13]. On the other hand, we propose a fuzzy decision model for battery recharging. This section will delve into the input and output linguistic variables, as well as the various fuzzy rules that will govern the decision-making process.

This section transcribes the work presented in [AISYS24; ASPAI24] and the work still under review in [ALGO24].

5.2.1 Fuzzy model

We propose a system designed from the literature in particular [dDD15], which aims to manage decisions regarding energy levels and tasks for a robot or autonomous agent. The primary goal is to decide whether the agent should recharge its energy or continue with its current task based on its energy level and distances to the target and energy source.

The description of the model will use fuzzy logic as described in [Fou13]. This approach relies on a formal knowledge representation using fuzzy sets, built on fuzzy elementary propositions of the form “ V is A ”. These propositions are defined from a set L (V , X , D_V) consisting of the linguistic variable V , the universe of possible values X , and a set of descriptions D_V for V , which are represented by fuzzy subsets of X .

Then, our proposed system uses three input variables and one output variable:

- **Energy level** - Input linguistic variable: *EnergyLevel*
- **Distance to target** - Input linguistic variable: *DistanceToTarget*
- **Distance to energy source** - Input linguistic variable: *DistanceToEnergySource*
- **Decision** (recharging or finishing the task) - Output linguistic variable: *Decision*

Each variable is defined over a specific range and described using linguistic terms, represented as fuzzy sets.

Input linguistic variables of the fuzzy decision model

The fuzzy decision model relies on three primary input variables to make intelligent decisions regarding energy management and task execution. These input variables are defined as follows:

- **EnergyLevel** - This variable represents the energy level of the system, measured as a percentage from 0 to 100. **Figure 5.1** illustrates the chosen membership functions for the input variable *EnergyLevel*. It is categorized into four linguistic terms:
 - *Empty*: [0, 20]
 - *Caution*: [10, 30]
 - *Operational*: [20, 75]
 - *Full*: [65, 100]
- **DistanceToTarget** - This variable measures the distance to the target in meters, ranging from 0 to 100. **Figure 5.2** depicted the chosen membership functions for the input variable *DistanceToTarget*. It is described using three linguistic terms:
 - *Near*: [0, 25]
 - *Medium*: [10, 40]
 - *Far*: [25, 100]
- **DistanceToEnergySource** - This variable indicates the distance to the energy source in meters, also ranging from 0 to 100. **Figure 5.3** shown the chosen membership functions for the input variable *DistanceToEnergySource*. It is classified into three linguistic terms:
 - *Near*: [0, 25]
 - *Medium*: [10, 40]
 - *Far*: [25, 100]

These variables form the foundation of the fuzzy decision-making process, allowing the system to interpret and respond to different scenarios based on the defined fuzzy rules.

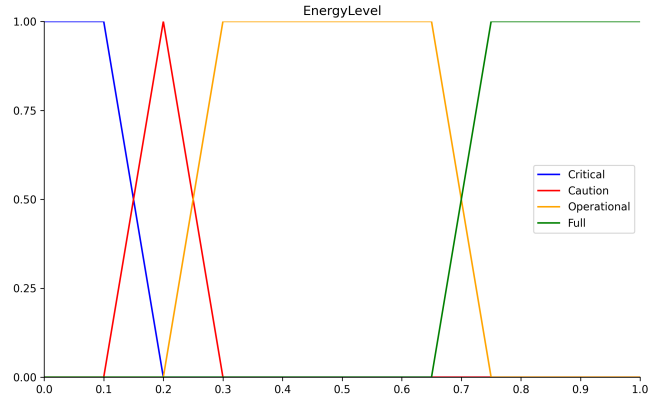


Figure 5.1 – Energy level

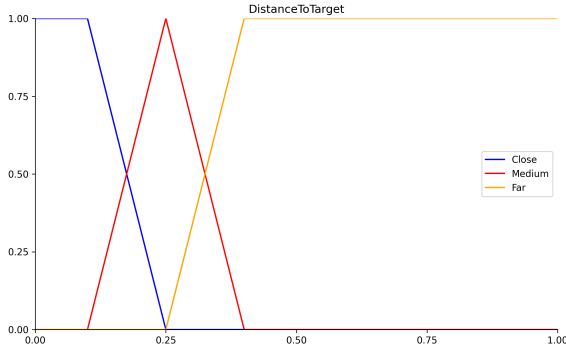


Figure 5.2 – Distance to target

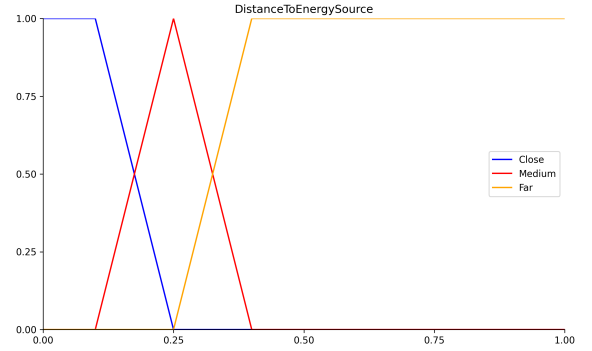


Figure 5.3 – Distance to energy source

Output linguistic variable of the fuzzy decision model

The fuzzy decision model produces one output variable that guides the system’s actions regarding energy management. This output variable is defined as follows:

- **Decision** - this variable ranges from 0 to 1 and determines whether the system should recharge or finish its current task. **Figure 5.4** depicted the chosen membership functions for the output variable *Decision*. It is described using two linguistic terms:
 - *Recharge*: $[0, 0.65]$
 - *FinishTask* $[0.325, 1]$

This output variable enables the system to make informed decisions based on the input variables and the fuzzy rules, ensuring optimal performance and energy efficiency.



Figure 5.4 – Decision

Fuzzy rules

The fuzzy rules extract from the literature [dDD15], establish the decision-making framework of the system. Each rule outlines conditions based on the input variables and determines the corresponding output decision. Our system adheres to the following rules:

1. If *EnergyLevel* is *Empty*, then *Decision* is *Recharge*.
2. If *EnergyLevel* is *Caution* and *DistanceToTarget* is *Near* and *DistanceToEnergySource* is *Far*, then *Decision* is *Recharge*.
3. If *EnergyLevel* is *Caution* and *DistanceToTarget* is *Near* and *DistanceToEnergySource* is *Medium*, then *Decision* is *FinishTask*.
4. If *EnergyLevel* is *Caution* and *DistanceToTarget* is *Near* and *DistanceToEnergySource* is *Near*, then *Decision* is *FinishTask*.
5. If *EnergyLevel* is *Caution* and *DistanceToTarget* is *Medium*, then *Decision* is *Recharge*.
6. If *EnergyLevel* is *Caution* and *DistanceToTarget* is *Far*, then *Decision* is *Recharge*.
7. If *EnergyLevel* is *Operational*, then *Decision* is *FinishTask*.
8. If *EnergyLevel* is *Full*, then *Decision* is *FinishTask*.

To find the final decision, the first step in the fuzzy inference process is fuzzification, where the system evaluates each elementary condition in the rule premises. This

involves taking the precise input values and mapping them to fuzzy values based on their membership in predefined fuzzy sets.

Once the input values are fuzzified, the system calculates the premises of each rule ("IF V is A "). The result of this calculation is known as the activation value of the rule, which indicates how strongly the rule is triggered by the current inputs. Then, the implication function is applied to the conclusion of each rule, producing a fuzzy subset that represents the distribution of possible output values based on the activated rule.

These fuzzy subsets, corresponding to the same output across different rules, are then combined using an aggregation method—often the maximum function. This step merges the outputs of all relevant rules into a single fuzzy subset.

Finally, the aggregated fuzzy subset is converted into a precise output value through a process called defuzzification. Among various defuzzification methods, one of the most common is the centroid or barycenter method, which calculates the center of gravity of the area under the curve representing the fuzzy subset. We will use this method in our simulations because it produces a single, clear output value that the system can use to make a final decision.

5.3 Fuzzy agent-based simulation for managing battery recharging

In the era of autonomous systems, effective battery management is crucial for the reliable operation of AIVs. This section explores a fuzzy agent-based simulation approach for managing battery recharging, focusing on a case study of autonomous battery recharging management in **subsection 5.3.1**. An adaptable fuzzy multi-agent model is presented and a multi-agent interface to address this problem.

We compare threshold-based models with fuzzy logic models, examining how fuzzy logic handles uncertainty and variability in **subsection 5.3.2**.

In **subsection 5.3.3**, the discussion advances to explore how increasing the complexity of fuzzy logic criteria can further enhance the system. Three key heuristics are introduced: (1) adapting recharging strategies based on demand and charging point availability, (2) adapting recharging according to the baggage arrival rate and (3) adjusting the speed of AIVs in response to the flow of baggage arrivals. These improvements demonstrate the potential of fuzzy logic to optimize the autonomous management of battery recharging in dynamic environments.

In addition, we have described the fuzzy agent model in the **appendix E**.

This section resulted from a publication on the ASPAI conference [ASP24] and the work under review [ALGO24].

5.3.1 Case study: autonomous management of battery recharging

We present an adaptable fuzzy multi-agent model **Figure 5.5** that addresses the challenges of energy management for AIVs. Efficient management of AIVs requires a holistic approach that takes into account several factors, including operational availability, energy consumption [Las+14], collaboration between AIVs and the dynamic infrastructure, and their adaptation to changing conditions. We aim to optimise recharging based on energy costs, as a low workload combined with frequent recharging can increase the overall energy consumption of the system. In addition, poor anticipation can limit system availability.

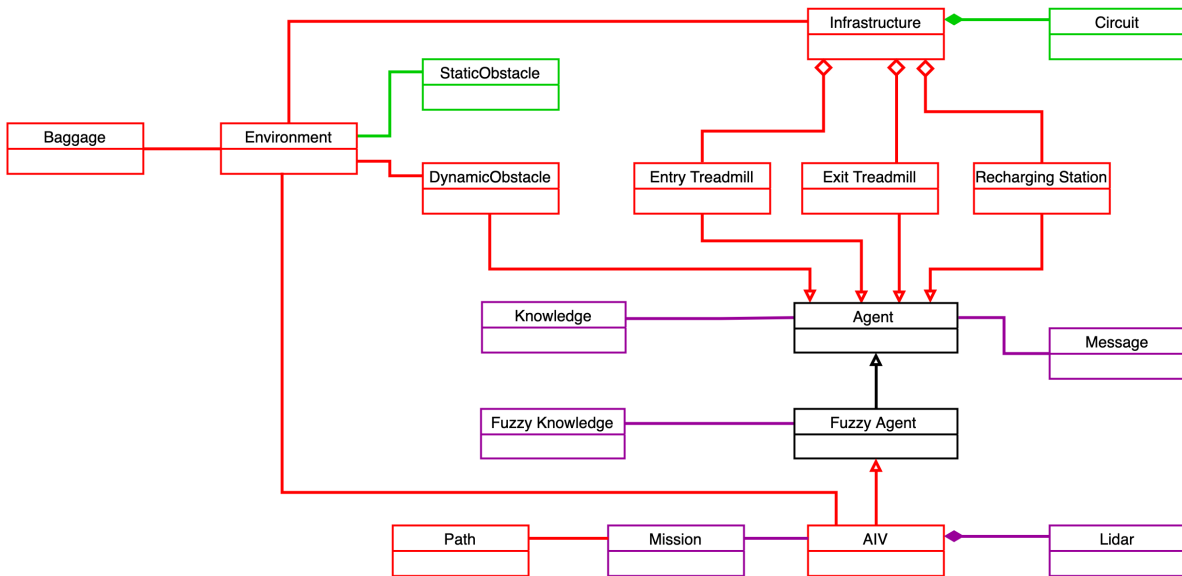


Figure 5.5 – Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple.

The interface of our “Airport Baggage Handling Simulation” application is depicted in **Figure 5.6**. We described the different functionalities of this interface simulation in **appendix B - section B.5**. This intuitive and structured interface allows us to monitor and analyze the performance of AIVs. The circulation scenario is detailed with a distance

oriented graph presented in **Figure 5.7**.

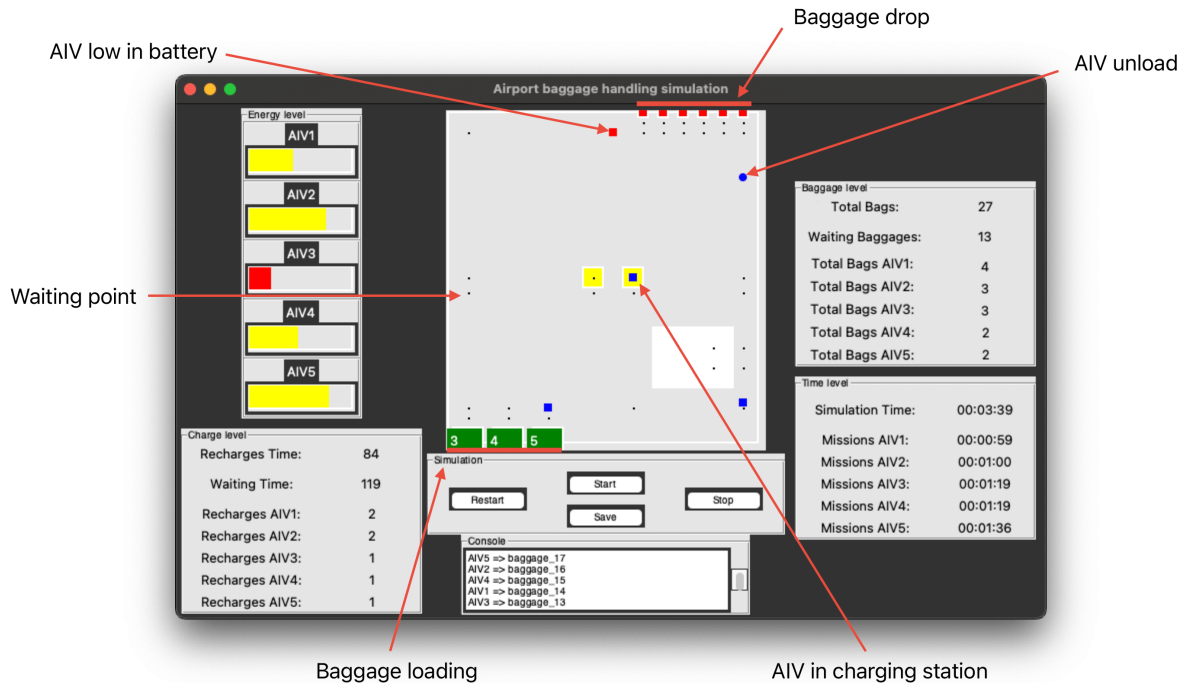


Figure 5.6 – Airport baggage handling simulation HMI

5.3.2 Comparisons between thresholds and fuzzy logic models

To test different autonomous management strategies for solving the problem of AIVs recharging batteries, we defined an initial circulation environment (**Figure 5.6**). We proposed different scenarios and compared them with the following four parameters:

- *nbMissions*: number of missions carried out.
- *timeMission*: the average time taken to complete a mission in seconds.
- *nbRecharges*: the number of recharges performed.
- *wtRecharging*: waiting times for recharging in seconds.

We also varied the charge threshold at which an AIV must recharge its battery. We then introduced a fuzzy inference system to determine the recharge time. We also adjusted the values of the fuzzy model (fuzzy linguistic values).

In this section, we delve into a comparative analysis between different thresholds and fuzzy logic models. We propose 3 different scenarios:

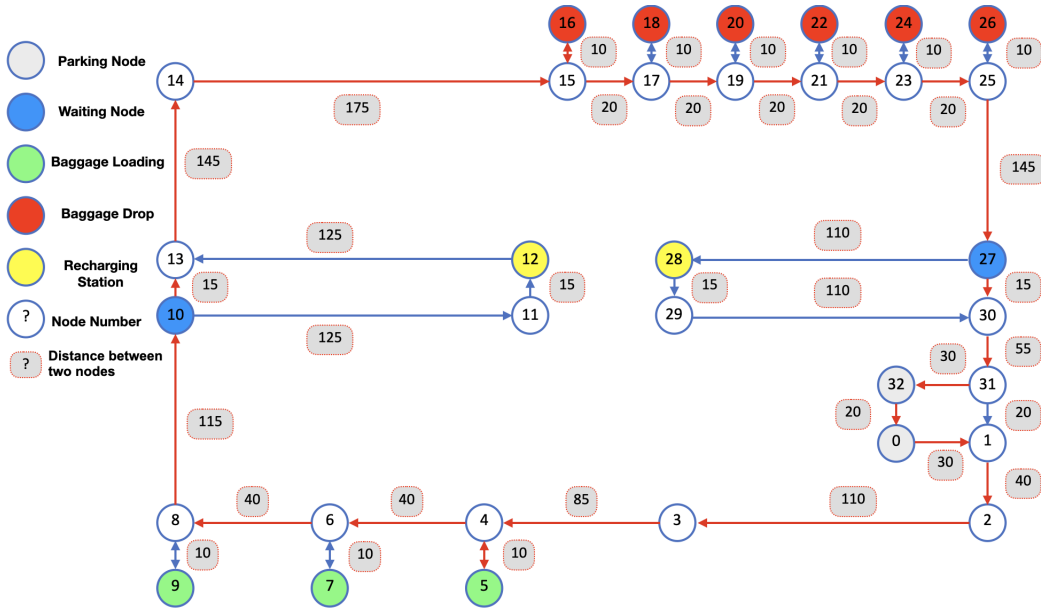


Figure 5.7 – Oriented graph: distance in the environment in meters

- Scenario 1 (Sc_1): all AIVs have a uniform recharge threshold of 30%.
- Scenario 2 (Sc_2): each AIV has a different recharge threshold, maintaining the same context as Sc_1 .
- Scenario 3 (Sc_3): AIVs use a fuzzy logic model for recharge.

We simulated these three scenarios for 1000 bags (a discussion regarding the scenario results is provided in the following three sections). The temporal results are shown in **Table 5.1**. We aim to discern the optimal threshold configurations that maximise mission throughput, minimise recharging frequency, and optimise resource utilisation, thereby improving the overall efficiency of autonomous management strategies for recharging the AIV battery.

Scenarios	Sc_1	Sc_2	Sc_3
Number of bags	1000	1000	1000
Total recharge time (s)	4800	4619	4345
Total simulation time	03:42:54	03:42:42	03:38:34

Table 5.1 – Time results for 1000 bags for Sc_1 , Sc_2 and Sc_3

Basic Scenario

In the *Basic Scenario*, AIVs have a single threshold model set at 30% for recharge. This scenario makes it possible to compare performance in terms of mission processing time (overall and individual time), number of recharges, and waiting time for recharges (access to a free station). The AIVs results for Sc_1 are shown in **Table 5.2**.

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
Thresholds	30	30	30	30	30	
<i>nbMissions</i>	201	199	200	200	200	1000
<i>timeMission</i>	64	64	64	64	64	64
<i>nbRecharges</i>	80	80	80	80	80	400
<i>wtRecharging</i>	0	93	42	68	93	244

Table 5.2 – Performance indicators for Sc_1

Different threshold values

Sc_2 enables us to compare different threshold values for AIVs recharge. Results are depicted in **Table 5.3**. When we compare with thresholds varying between 15% and 30%, the overall mission processing time is slightly lower, and the number of recharges and overall recharge time are also lower (374 and 400, respectively). The performance of AIV₁ with the lowest threshold (15%) is obviously the best for the average time taken to complete a mission time. However, there is a greater risk of not being able to reach a station due to a lack of charge in the event of an incident!

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
Thresholds	15	20	25	30	35	
<i>nbMissions</i>	202	201	199	199	202	1000
<i>timeMission</i>	63	64	64	64	64	63.8
<i>nbRecharges</i>	67	67	80	80	80	374
<i>wtRecharging</i>	180	140	49	77	51	497

Table 5.3 – Performance indicators for Sc_2

Fuzzy logic model

In comparison with Sc_1 , where AIVs have a threshold of 30%, in Sc_3 , AIVs use a fuzzy basic model. The results are presented in **Table 5.4** demonstrate an improvement

in overall and individual AIV times (63 secondes on average instead of 64 secondes) and fewer recharges (335 recharges instead of 400).

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
FL model	FL	FL	FL	FL	FL	
<i>nbMissions</i>	201	200	200	200	199	1000
<i>timeMission</i>	63	63	63	63	63	63
<i>nbRecharges</i>	67	67	67	67	67	335
<i>wtRecharging</i>	0	58	19	49	71	197

Table 5.4 – Performance indicators for Sc_3

5.3.3 Increases in fuzzy logic criteria

To improve the results of the previous simulations, we made three types of adaptation (heuristics) corresponding to three new scenarios, taking into account more realistic constraints and the possibility of AIVs communicating with each other and with infrastructure elements such as charging stations:

1. Sc_4 : adaptation of recharging according to the needs of the AIVs and the availability of the charging points (centralized scenario by supervision and decentralized scenario by communication between the AIVs and the charging points);
2. Sc_5 : adaptation of recharging according to the rate of baggage arrival and the resulting variation in activity (the number of missions to be performed by the AIVs in a unit of time is no longer constant);
3. Sc_6 : adapting the speed of the AIVs according to the rate of baggage arrival (centralized scenario by supervision and decentralized scenario by communication between the AIVs and the charging points).

The objective of this section is to show that specific heuristics allow certain situations to be dealt with fairly finely and increase the collective/overall performance of AIVs. We simulated these three improved scenarios for 1000 bags. The temporal results are shown in **Table 5.5**.

Adapting recharging to demand and the availability of charging points

The first heuristic, referred to as Sc_4 , simulates the adaptation of charging behaviour to both demand and the availability of charging points. The AIV results are presented

Scenarios	Sc ₄	Sc ₅	Sc ₆
Number of bags	1000	1000	1000
Total recharge time (s)	4606	4285	935
Total simulation time	03:39:08	03:37:03	01:32:15
Maximum number waiting bags	468	659	159
Average bags waiting	234	327	99,62

Table 5.5 – Time results and configuration for 1000 bags for Sc_4 , Sc_5 and Sc_6

in **Table 5.6**. The effectiveness of this heuristic is obvious, particularly for AIV_1 , which required 14 fewer recharges compared to AIV_5 and 12 fewer recharges than AIV_4 . Additionally, the total recharging time for Sc_4 is shorter than for both Sc_1 and Sc_2 : 4606 seconds compared to 4800 seconds and 4619 seconds, (respectively **Table 5.5** for Sc_4 and **Table 5.1** for Sc_1 and Sc_2).

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
Thresholds	15/15	20/15	25/20	30/20	35/25	
<i>nbMissions</i>	203	201	199	198	199	1000
<i>timeMission</i>	62	62	63	63	63	62.6
<i>nbRecharges</i>	68	67	78	76	80	369
<i>wtRecharging</i>	143	178	5	10	70	406

Table 5.6 – Performance indications for Sc_4

Adaptation of recharging according to the baggage arrival rate

The second heuristic, referred to as Sc_5 , simulates recharging adaptations based on the baggage arrival rate and the corresponding variation in induced activity (the number of tasks to be performed by the AIVs). As shown in **Table 5.7**, this heuristic enables the AIVs to complete their missions more quickly compared to Sc_4 . Specifically, AIVs complete one mission in an average of 58 seconds under Sc_5 , whereas it takes them 62.6 seconds under Sc_4 (see **Table 5.6**).

Adapting the speed of the AIVs to the flow of baggage arrivals

The final heuristic, referred to as Sc_6 , adapts the speed of the AIVs to the flow of baggage arrivals. Compared to Sc_5 , the 30% speed threshold has been adjusted, as the 20% threshold led to too many load faults due to increased energy consumption at higher

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
Thresholds	20	20	20	20	20	
<i>nbMissions</i>	200	199	201	200	200	1000
<i>timeMission</i>	58	58	58	58	58	58
<i>nbRecharges</i>	15	15	15	15	15	75
<i>wtRecharging</i>	66	0	25	57	94	242

Table 5.7 – Performance indicators for Sc_5

speeds. This adjustment results in a significantly shorter overall simulation time, as shown in **Table 5.5**. Additionally, **Table 5.8** demonstrates improved throughput control, with bags waiting times reduced to 99.6 seconds in this scenario, compared to 327 seconds for Sc_5 (see **Table 5.7**).

Indicators	AIV ₁	AIV ₂	AIV ₃	AIV ₄	AIV ₅	Global
Thresholds	30	30	30	30	30	
<i>nbMissions</i>	201	199	200	199	201	1000
<i>timeMission</i>	25	25	25	25	25	25
<i>nbRecharges</i>	17	17	17	17	17	85
<i>wtRecharging</i>	0	19	76	20	54	169

Table 5.8 – Performance indicators for Sc_6

5.4 Problem of recharging batteries with TA

As mentioned in the introduction of this chapter, AIV missions do not follow a uniform distribution in terms of frequency, creating periods of intense activity and others that are quieter. It is therefore essential to link the energy consumption of AIVs to the amount of work carried out and their operational availability.

Efficient TA remains a critical challenge, as discussed in **chapter 3**, especially in dynamic environments where multiple AIVs are tasked with handling various missions. One of the primary objectives is to develop strategies that ensure optimal TA, thereby maximizing the overall efficiency and effectiveness of the fleet.

In this section, we simulate different scenarios within the same multi-agent application context as in **section 5.3**.

We begin with an analysis of TA using an auction-based approach grounded in fuzzy inference. It corresponds to the first simulation, which we will explore in **subsection**

5.4.1. Fuzzy logic enhances our ability to understand natural, uncertain, and imprecise phenomena by utilizing rules and membership functions to define “fuzzy sets”. This approach effectively addresses the uncertainties and imprecisions inherent in AIV operations. By modelling human reasoning and managing imprecise information, fuzzy logic provides a robust framework for decision-making in complex and uncertain environments.

Following this, we will simulate a scenario using the fuzzy decision model for TA within the auction strategy and incorporate the model outlined in **subsection 5.2.1** for recharging management. Finally, we will enhance the fuzzy recharging model to develop a more realistic and effective version.

This section is linked to the contribution [[AISYS24](#)].

5.4.1 Basic TA auction strategy with fuzzy rules

We propose a fuzzy system inspired by the literature [[AB21](#); [XZ24](#)] for basic TA auction strategy, referred as *FuzzyBasicTa*, designed to incorporate fuzzy rules that handle the uncertainties and variabilities associated with AIV operations. In the interface of our “Airport Baggage Handling Simulation” application, simulating this strategy corresponds to selecting scenario 7: Sc_7 (refer to **appendix B - section B.5**). Fuzzy logic provides a robust framework for decision-making in complex and uncertain environments, enabling more effective TA.

Sc_7 implements *FuzzyBasicTa* a TA strategy in which each AIV agent uses a fuzzy model with three input variables and one output variable:

- **Availability** (in relation to the number of bags the AIV must handle) - Input linguistic variable: *Availability*
- **Distance to target** (distance from the baggage drop-off location) - Input linguistic variable: *DistanceToTarget*
- **Level of energy** - Input linguistic variable: *EnergyLevel*
- **Cost** (cost of the task) - Output linguistic variable: *Cost*

Each variable is defined over a specific range and described using linguistic terms, represented as fuzzy sets.

Input linguistic variables of the TA fuzzy decision model

- **Availability** - This variable represents the number of bags the AIV must handle, measured as a number from 0 to 20. It is categorized into three linguistic terms:

- *Weak*: [0, 2]
- *Medium*: [1, 3]
- *High*: [2, 20]
- **Distance to target** - This variable measures the distance to the target (from the baggage drop-off location) in meters, ranging from 0 to 100. *DistanceToTarget*. It is described using three linguistic terms:
 - *Near*: [0, 40]
 - *Medium*: [10, 80]
 - *Far*: [50, 100]
- **Level of energy** - This variable indicates the energy level of the system, measured as a percentage from 0 to 100. It is classified into four linguistic terms:
 - *Empty*: [0, 20]
 - *Caution*: [10, 30]
 - *Operational*: [20, 75]
 - *Full*: [75, 100]

Output linguistic variable of the TA fuzzy decision model

The fuzzy decision model produces one output variable that guides the system's actions regarding TA. This output variable is defined as follows:

- **Cost** - This variable ranges from 0 to 1 and determines whether the system should recharge or finish its current task. It is described using five linguistic terms:
 - *VeryWeak*: [0, 0.2]
 - *Weak*: [0.1, 0.5]
 - *Medium*: [0.3, 0.7]
 - *High*: [0.5, 0.9]
 - *VeryHigh*: [0.8, 1.0]

This output variable enables the system to make informed decisions based on the input variables and the fuzzy rules, ensuring optimal performance and energy efficiency.

Fuzzy rules of the TA fuzzy decision model

The fuzzy rules establish the decision-making framework of the system. Each rule outlines conditions based on the input variables and determines the corresponding output decision. Our system adheres to the following rules:

1. If *Availability* is *Weak* then *Cost* is *VeryHigh*.
2. If *Availability* is *Weak*, *DistanceToTarget* is *Near* and *EnergyLevel* is *Full*, then *Cost* is *High*.
3. If *Availability* is *Medium*, *DistanceToTarget* is *Far* and *EnergyLevel* is *Caution*, then *Cost* is *High*.
4. If *Availability* is *Medium*, *DistanceToTarget* is *Medium* and *EnergyLevel* is *Operational*, then *Cost* is *Medium*.
5. If *Availability* is *Medium*, *DistanceToTarget* is *Near* and *EnergyLevel* is *Full*, then *Cost* is *Weak*.
6. If *Availability* is *Medium*, *DistanceToTarget* is *Far* and *EnergyLevel* is *Caution*, then *Cost* is *Weak*.
7. If *Availability* is *High*, then *Cost* is *VeryWeak*.

Results with the *FuzzyBasicTA* strategy

The results of this new strategy *FuzzyBasicTa*, in **Figure 5.9** are promising: the maximum number of bags waiting is low, the overall simulation time is efficient, the missions are well-distributed among the AIVs, and the average AIV occupancy rate is high at 0.88.

However, there are a few elements of uncertainty in this scenario, defined by three linguistic variables. Fuzzy logic helps to better understand these uncertain, imprecise, and difficult-to-model natural phenomena by defining rules and membership functions for “fuzzy” sets. Introducing additional fuzzy elements, such as nuanced simulation parameters, is expected to improve results, particularly by reducing the maximum number of waiting bags items and enhancing battery recharging management.

5.4.2 Energy management with fuzzy rules

In this subsection, we introduce Scenario 8 (Sc_8), which extends the strategy of the fuzzy model discussed in the following subsection by incorporating an additional layer of energy management through a second fuzzy model. This enhanced model enables AIVs to assess whether they need to recharge during a mission, thereby refining the calculation of the overall mission cost.

Strategy / Parameters	<i>FuzzyBasicTa</i>
Maximum number of bags waiting	6
Overall simulation time (in s)	1843
Average mission time per AIV (in s)	[80, 81, 80, 81, 82]
Number of missions carried out by AIV	[21, 21, 21, 19, 18]
Occupancy rate by AIV	[0.91, 0.92, 0.91, 0.84, 0.80]
Average occupancy rate	0.88

Table 5.9 – Global results of the strategy *FuzzyBasicTa*

Sc_8 employs the *FuzzyBasicTaAndRecharge* strategy, integrating the new fuzzy model for recharging. This model is characterized by five input variables and one output variable:

- **Level of energy** - Input linguistic variable: *EnergyLevel*
- **Distance to station 1** - Input linguistic variable: *DistanceToStation1*
- **Distance to station 2** - Input linguistic variable: *DistanceToStation2*
- **Availability station 1** - Input linguistic variable: *AvailabilityStation1*
- **Availability station 2** - Input linguistic variable: *AvailabilityStation2*
- **Recharge** - Output linguistic variable: *Recharge*

The *EnergyLevel* variable is defined using the same specific range and linguistic terms as in the *FuzzyBasicTa* strategy detailed in the subsection above.

Input linguistic variables of the recharge fuzzy decision model

- **Distance to station 1** or **Distance to station 2** - This variable represents the distance to the respective station that the AIV must consider, measured on a scale from 0 to 1200 (in meters). It is categorized into three linguistic terms:
 - *Near*: [0, 600]
 - *Medium*: [0, 1200]
 - *Far*: [600, 1200]

- **Availability station 1** or **Availability station 2** - This variable represents the distance to the station (1 or 2) the AIV must handle, measured as a number from 0 to 5 (corresponding to the number of AIVs in the simulation). It is categorized into three linguistic terms:
 - *High*: [0, 1]
 - *Medium*: [0, 2]
 - *Weak*: [1, 5]

Output linguistic variable of the recharge fuzzy decision model

The fuzzy decision model produces one output variable that decides if the AIV has to finish its task and go to recharge at *Station1* or *Station2*. This output variable is defined as follows:

- **Recharge** - This variable ranges from 0 to 1 and dictates whether the system should recharge or continue its current task. It is described using three linguistic terms:
 - *Station1*: [0, 0.3]
 - *Station2*: [0.1, 0.5]
 - *FinishTask*: [0.4, 1.0]

We recall that the objective is to decide whether the AIV should recharge its energy or continue with its current task based on its energy level and distances to the target and energy source.

Results with the *FuzzyBasicTaAndRecharge* strategy

In the new scenario where AIV employs the *FuzzyBasicTaAndRecharge* strategy (scenario 8, denoted as Sc_8), the performance outcomes for TA are marginally less favorable compared to those in scenario Sc_7 . Detailed results for Sc_8 are shown in **Tables 5.10** and **5.11**.

Indeed, the maximum number of bags waiting remains the same, the overall simulation time is slightly longer, the distribution of missions among the AIVs is less efficient, and the average AIV occupancy rate is lower at 0.82. However, the overall recharging time is reduced in this scenario, potentially leading to greater AIV availability. This represents an area for improvement in future scenarios.

Strategy / Parameters	<i>FuzzyBasicTaAndRecharge</i>
Maximum number of bags waiting	6
Overall simulation time (in s)	2000
Average mission time per AIV (in s)	[81, 80, 81, 84, 83]
Number of missions carried out by AIV	[23, 19, 21, 19, 18]
Occupancy rate by AIV	[0.93, 0.76, 0.85, 0.80, 0.75]
Average occupancy rate	0.82

Table 5.10 – Global results of the strategy *FuzzyBasicTaAndRecharge*

Strategies / Parameters	<i>FuzzyBasicTaAndRecharge</i>
Total recharge time	490
Total waiting time for refills	124
Total number of refills	33
Breakdown of the number of recharges by AIV	[8, 6, 7, 6, 6]

Table 5.11 – Results linked to recharges with the *FuzzyBasicTaAndRecharge* scenario

5.4.3 Improving the fuzzy recharge model

In this scenario entitled *FuzzyBasicTaAndImprovedRecharge*, we propose a more realistic version of the fuzzy model presented in **subsection 5.2.1** and used in the subsequent simulation. The system’s objective remains the same: to decide whether the agent should recharge its energy or continue with its current task based on its energy level and distances to the target and energy source.

To enhance the model, we introduce additional input variables. The improved model now considers urgency in relation to the number of baggage awaiting delivery (speed), proximity with other AIVs, alongside existing factors such as energy level, distance to

recharging stations, and station availability.

New input linguistic variables of the improved recharge fuzzy decision model

The system now also incorporates the following new linguistic variables:

- **Urgency** - This variable represents the distance to the station (1 or 2) the AIV must handle, measured as a number from 0 to 10. It is categorized into three linguistic terms:
 - *Weak*: [0, 3]
 - *Medium*: [0, 5]
 - *High*: [3, 10]
- **Proximity AIV** - This variable represents the proximity of other AIV agents (use of observed and safety distances), measured as a number from 0 to 1000 (meters). It is categorized into three linguistic terms:
 - *High*: [0, 300]
 - *Medium*: [100, 500]
 - *Weak*: [300, 1000]

Output linguistic variable of the improved recharge fuzzy decision model

The improved fuzzy decision model introduces a new output variable, *Speed*, and refines the existing output variable, *Recharge*, to be more precise. These output variables are defined as follows:

- **Speed** - This variable represents the speed of the AIV, measured in meters per second (m/s) within the range of 0 to 10. It is categorized into three linguistic terms:
 - *Weak*: [0, 5]
 - *Normal*: [2, 8]
 - *High*: [5, 10]
- **Recharge** - This variable ranges from 0 to 1 and dictates whether the system should recharge or continue its current task. It is described using four linguistic terms:
 - *Speed*: [0.4, 0.8]
 - *Normal*: [0.6, 1.0]
 - *Strong*: [0.8, 1.0]
 - *FinishTask*: [0, 0.6]

Results with the *FuzzyBasicTaAndImprovedRecharge* strategy

This enriched model aims to refine the decision-making process for when and how quickly an AIV should recharge, balancing the urgency of baggage deliveries with efficient energy management. By integrating these heuristics, the objective is to minimize the waiting time for a recharge while ensuring AIVs are readily available to pick up and deliver baggage promptly. This approach seeks to adjust the speed of AIVs based on the real-time flow of baggage arrivals, aiming to reduce the maximum number of waiting for bags and thereby improve overall operational efficiency and availability of AIVs for continuous luggage handling tasks.

The comparison between the two scenarios, *FuzzyBasicTaAndRecharge* (Sc_8) and *FuzzyBasicTaAndImprovedRecharge* (Sc_9), reveals significant differences in their performance metrics.

The global results presented, in **Figure 5.12**, demonstrate the efficacy of the improved model. Sc_9 exhibits a shorter overall simulation time of 1675 seconds compared to 2000 seconds in Sc_8 , indicating more efficient operations. The average mission time per AIV is also significantly reduced in Sc_9 , ranging from 65 to 67 seconds, compared to 80 to 84 seconds in Sc_8 , reflecting more effective task handling. Mission counts are fairly similar across both scenarios. The occupancy rates in Sc_9 , averaging at 0.79, are slightly lower than the 0.82 average in Sc_8 . This lower rate suggests more efficient utilization of AIVs.

Recharge metrics further highlight the improvements with the use of *FuzzyBasicTaAndImprovedRecharge*, depicted in **Table 5.13**. Despite a higher total recharge time of 736 seconds compared to 490 seconds in Sc_8 . Sc_9 features a more efficient distribution of recharge events, with 49 total refills compared to 33 in Sc_8 . The increase in the speeds of the AIV agents has an energy cost! The waiting time for refills is slightly lower in Sc_9 at 119 seconds compared to 124 seconds in Sc_8 . The breakdown of recharges by AIV in Sc_9 (ranging from 7 to 11) also indicates a more balanced and optimized recharge strategy compared to Sc_8 (ranging from 6 to 8).

In summary, the use of the strategy *FuzzyBasicTaAndImprovedRecharge* demonstrates a superior approach to TA and recharge management, leading to better performance metrics across various parameters compared to the strategy *FuzzyBasicTaAndRecharge*. This enhanced model effectively balances the urgency of baggage deliveries with efficient energy management, resulting in improved operational efficiency and availability of AIVs for continuous baggage handling tasks.

Strategy / Parameters	<i>FuzzyBasicTaAndImprovedRecharge</i>
Maximum number of bags waiting	6
Overall simulation time (in s)	1675
Average mission time per AIV (in s)	[67, 65, 67, 65, 67]
Number of missions carried out by AIV	[22, 22, 22, 19, 15]
Occupancy rate by AIV	[0.88, 0.85, 0.88 0.74, 0.6]
Average occupancy rate	0.79

Table 5.12 – Global results of the strategy *FuzzyBasicTaAndImprovedRecharge*

Strategy / Parameters	<i>FuzzyBasicTaAndImprovedRecharge</i>
Total recharge time	736
Total waiting time for refills	119
Total number of refills	49
Breakdown of the number of recharges by AIV	[11, 11, 11, 9, 7]

Table 5.13 – Results linked to recharges with the *FuzzyBasicTaAndImprovedRecharge* scenario

5.5 Conclusion

In this chapter, we have proposed a multi-agent simulation (**appendix B - section B.5**), including fuzzy logic, to test various scenarios of battery recharging management for mobile baggage conveyor robots (AIVs) at an airport. This approach offers a flexible adaptation to the various aspects of AIV management and facilitates possible adjustments needed for deployment at an airport site.

The use of a distributed multi-agent system provides temporary autonomy in case of central infrastructure failure and can improve the management of individual AIV functions, such as TA, battery charging, speed regulation, etc. Our main simulation objective was to minimize the maximum number of bags waiting at a given time, the total simulation time, the average time to complete a mission per AIV, the number of missions completed per AIV during the simulation, and the activity rate of the AIVs.

In **section 5.3**, the simulation results demonstrate that incorporating adaptive fuzzy multi-agent models (**section 5.2**) for AIV energy management can significantly optimize recharging strategies, improve operational efficiency, and mitigate energy consumption, particularly by considering dynamic factors such as workload variation, communication between AIVs and infrastructure elements. In fact, an infrastructure capable of optimising recharging according to energy tariffs is advantageous, particularly with the ability to cut consumption over an hour. These findings will underscore the importance of flexible, collaborative approaches in enhancing the performance of autonomous systems in dynamic environments.

In **appendix D**, an optimal control problem is defined to improve an accurate energy consumption model. Subsequently, an algorithm is implemented to plot a speed profile based on the distance travelled and the corresponding energy profile, which considers two distances: acceleration and deceleration. Using this algorithm, three strategies are assessed. The strategy selected for the robot will be determined using a fuzzy logic model. During a robot mission, two constraints are imposed: waiting baggage and traffic, which are considered inputs to our fuzzy logic model for strategy selection.

Three strategies generated by our methodology effectively reflect these constraints. The goal is to find an optimal compromise between energy consumption, bags waiting times, and traffic conditions. As shown by numerical simulations, *Strategy1* focuses on optimal energy consumption. *Strategy2* allows for a slight increase in energy usage to address waiting baggage. *Strategy3* prioritizes alleviating energy constraints related to traffic while addressing waiting baggage.

In **section 5.4**, we explored an initial TA scenario using a basic fuzzy model. We then enhanced this scenario by refining the fuzzy decision model of the TA in several key areas: (1) optimizing the recharging of autonomous vehicle (TA) batteries, (2) selecting the most suitable recharging stations, (3) determining the optimal recharging rate, and (4) adjusting the TA speeds to match variations in baggage arrival rates.

The simulation results demonstrate that employing adaptive fuzzy multi-agent models

to manage TA, energy recharging, infrastructure selection, and speed adjustments significantly boost the operational efficiency of the AIV fleet. These strategies, which emphasize energy optimization and resource allocation, are directly aligned with the goals of "Level 4" on the collective autonomy scale (mentioned in **subsection 1.3.3**), where the focus shifts from individual task efficiency to collective optimization across the entire fleet. By reducing energy consumption and improving operational coordination, the system is better equipped to achieve sustainable and efficient performance in complex, dynamic environments.

We plan to enhance the behaviour simulations of our AIV agents by integrating fuzzy models, aiming to improve the relevance and effectiveness of their decision-making in collective autonomy management. Additionally, we intend to develop strategies that better address various constraints, such as prioritizing fully loaded robots over empty ones and taking into account the battery levels of the robots.

An important consideration for future research is the heterogeneity of the AIVs, as these robots may possess different and discriminating characteristics that impact TA and overall system performance. By considering the diverse capabilities, battery capacities, and operational efficiencies of heterogeneous AIVs, we can further refine our fuzzy logic models to ensure more tailored and efficient mission planning and execution. This approach will better align the deployment of AIVs with their specific strengths and limitations, ultimately enhancing the system's overall efficiency and resilience.

CONCLUSION

This thesis set out to explore and enhance the autonomy of AIVs within industrial fleets by applying collective intelligence strategies. Focusing on the core technical functions of AIVs: obstacle avoidance, Task Allocation (TA), cooperative perception, and energy management, this work offers a comprehensive investigation into how collective strategies can improve both individual and fleet-wide performance.

The objective of this thesis was to propose and evaluate strategies that increase the autonomy and operational efficiency of AIV fleets by enabling robust communication, cooperation, and decision-making among vehicles. This goal was driven by the understanding that future industrial systems will increasingly rely on fleets of heterogeneous robots working collaboratively to perform complex tasks. By enhancing the collective intelligence of these vehicles, it becomes possible to tackle issues related to navigation in dynamic environments, TA under real-time constraints, and energy management across entire fleets. Given this context, one of the central questions posed in the introduction was:

What methodologies can be used to rigorously assess and refine collective strategies aimed at increasing the autonomy of AIV fleets?

Throughout this thesis, we employed a methodology that leverages multi-agent systems, simulation environments, and scenario-based testing, which proved to be essential tools for evaluating and refining collective strategies for AIV autonomy.

By decomposing the autonomy problem into manageable technical tasks — such as navigation, perception, TA, and energy management — we identified the key areas where collective strategies can be most effective. We have shown that the integration of adaptive algorithms within these strategies enables AIVs to make more autonomous decisions while facilitating real-time communication and information sharing between agents.

The proposed solutions were rigorously assessed through modeling and simulation, allowing us to validate their performance under dynamic and complex conditions. These tests not only ensured the practical viability of the strategies but also highlighted areas for further optimization. Decentralized decision-making proved particularly effective in handling the complexity of AIV systems, while scenario-based simulations allowed us to

stress-test these strategies in realistic operational environments.

In conclusion, the methodologies outlined in this thesis provide a robust framework for systematically assessing and refining collective strategies to enhance the autonomy of AIV fleets before moving on to real-time field experiments. By addressing the inherent complexity of these systems and ensuring that proposed solutions are tested, modeled, and validated, we have developed different multi-agent simulations (**appendix B**) that contribute to validating our different collective strategies.

Contributions

How can the navigation and localization capabilities of AIVs be enhanced in complex, dynamic environments through cooperative perception and collective strategies?

In the context of Industry 4.0, enhancing the navigation and localization capabilities of AIVs is vital for effective obstacle avoidance in complex, dynamic environments. Indeed, for example, warehouses are characterized by a variety of interacting elements, including vehicles, operators, and potential obstacles like objects that may fall or be left in aisles. This manuscript investigates the impact of cooperative perception among fleet vehicles and the application of collective strategies to optimize their operational efficiency during missions.

A key contribution is the extension of the message-based communication protocol introduced by Bahnes et al. in [BKH16], which originally prioritized vehicle passage through intersections. As a first step, we carried out an algorithmic work to extend this algorithm in order to have the possibility to manage the detection of fixed and mobile obstacles (**Figure 2.1**) [ARCI22].

By integrating a cooperative perception protocol based on the ETSI standard for ITS, vehicles now share location data, environmental perception, and warnings of hazardous events through standard messages. This framework is validated through simulation, demonstrating how AIVs in a fleet can coordinate more effectively when responding to evolving obstacles in real time.

Moving toward a collective strategy for global obstacle avoidance (in **section 2.3**), our simulation results demonstrated the benefits of collaboration between AIVs, showing that both individual and fleet efficiency increased [ASPAI22]. This collaboration opens the possibility for future advancements, such as real-time TA, scheduling, and redistribution among AIVs in response to perceived obstacles. Furthermore, incorporating cooperative

perception, either through AIV-based tracking or infrastructure support, could further enhance fleet efficiency during mission execution.

Finally, as discussed in **appendix C**, we developed a method for estimating AIV positions within a closed industrial environment [ICAE23] and extended a collision detection algorithm for more robust obstacle avoidance [ASPAI22]. These methods were validated using an agent-based simulation platform that integrates both obstacle detection and position estimation algorithms.

After the different works about obstacle avoidance, our aim was to move to a higher level of collaboration with the possibility of mission sharing and reorganization in a distributed way.

What strategies can optimize TA within a fleet of AIVs to enhance operational efficiency and reduce downtime?

In the context of the current smart factory, AIVs must become increasingly autonomous in order to perform their missions effectively, i.e. optimize their activity according to performance indicators such as distances covered, energy consumed, time for performing missions, availability, etc. Autonomy and decentralization are two excessively linked notions to the extent that an autonomous system operates and makes decisions autonomously.

Therefore, we proposed a dynamic task (re-)allocation process model for AIVs, managing their activity in a decentralized context [ICAE24]. We then developed a multi-agent simulation (**appendix B - B.4**) to be able to simulate this process and test it on different scenarios of problematic traffic situations. The proposed scenarios allow us to move towards strong cooperation between AIV agents, but also between AIV agents and infrastructure agents (cameras). The V2X communication implemented to enable this cooperation is an essential element of our decentralized agent-based simulation approach. We have shown that it brings more flexibility and robustness in the management of problematic dynamic situations.

How can energy management be optimized across a fleet of AIVs to extend operational autonomy and reduce interruptions due to recharging or battery replacement?

Efficient management of AIVs requires a comprehensive approach considering several factors, including operational availability, energy consumption, collaboration between AIVs and infrastructure, as well as their adaptation to changing conditions.

Optimizing energy management across a fleet of AIVs can significantly extend their operational autonomy and reduce interruptions due to recharging or battery replacement. Our research demonstrates that our multi-agent simulation (**appendix B - B.5**), incorporating fuzzy logic, provides an effective framework for this optimization. By leveraging adaptive fuzzy multi-agent models, we can refine recharging strategies, enhance operational efficiency, and minimize energy consumption.

The simulation results highlight several key findings. Firstly, the integration of fuzzy logic into the energy management system allows for dynamic adjustments based on real-time factors such as workload variations and energy tariffs. This flexibility in recharging management enables AIVs to operate more efficiently by optimizing when and where they recharge [[ALGO24](#); [ASPAI24](#)].

Additionally, our approach to task adaptation and speed regulation — where AIVs adjust their speed in response to varying baggage arrival flows — further optimizes energy use. By coordinating these factors with recharging strategies, we ensure that the AIV fleet remains operational with minimal interruptions [[AISYS24](#)].

Furthermore, the use of a distributed multi-agent system enhances resilience by allowing individual AIVs to maintain autonomy in the event of central infrastructure failures. This capability ensures continuous operation and efficient management of each vehicle's energy needs.

Overall, our research underscores the importance of employing flexible, collaborative approaches in managing autonomous systems. The adaptive fuzzy multi-agent models not only improve recharging strategies and operational efficiency but also demonstrate the potential for more sustainable and effective management of AIV fleets in dynamic environments.

How can the heterogeneity of industrial systems be managed more effectively?

Managing the heterogeneity of industrial systems more effectively requires a comprehensive communication framework that ensures seamless interaction between AIVs and infrastructure. Our contributions demonstrate that leveraging ITS technologies, such as CAM, DENM, and CPM, adapted for industrial contexts [[ARAC24](#)], can significantly enhance communication reliability and efficiency.

Through our work during my mobility at the University of Tokyo, we developed a novel architecture for the Carla simulator, integrating OpenCDA and ROS2 to establish a V2X communication system [[VTC24](#)]. This architecture facilitates seamless interactions

between CAV and RSU, creating a simulated environment to generate V2X datasets and refine algorithms for ADAS. This system not only supports efficient management of communication heterogeneity but also enhances decision-making in complex scenarios, as demonstrated by our blind spot detection use case.

By integrating AIVs into the Carla simulator and industrial environment scenarios, this architecture can help work on V2Xs communication between AIVs and the infrastructure to improve decision-making. The ability to simulate various conditions and optimize V2X communications paves the way for more robust industrial operations, reduced inefficiencies, and improved interoperability of systems in dynamic environments.

Future works

In this manuscript, we have explored various strategies to enhance the core functionalities of AIVs through collective strategies, focusing on their interaction with both each other and the surrounding infrastructure. Building on these findings, the different subsections below present several areas promising avenues for further research and development as a result of the work carried out during this thesis.

Obstacle avoidance

A key aspect of this work will be to implement the Bahnes algorithm [BKH16] for real-world experiments. Additionally, there will be a push towards real-world experimentation, involving the use of physical robots to test the proposed strategies. This includes experimenting with Turtlebot3 "burger" robots and integrating intersection proximity infrastructure to optimize communication and coordination among AIVs. The aim is to reduce unnecessary braking and stopping, thereby improving energy efficiency and overall mission performance.

Another important direction is to expand the performance analysis by introducing new indicators and testing various mission scenarios with multiple robots and obstacles. This will help in evaluating the effectiveness of the collective strategies under diverse conditions.

Lastly, future work will explore further development of cooperative infrastructure and the application of fuzzy logic to better qualify and respond to obstacles. This will enhance the AIV agents' ability to navigate and operate efficiently in complex environments.

Task allocation

Our work highlights two key perspectives: data fusion and shared memory among AIV agents. These concepts are essential for improving cooperation between AIVs and other agents, such as cameras, to enhance decision-making and efficiency.

Data fusion involves integrating information from different sources, such as an AIV detecting an obstacle at one time and a camera detecting the same obstacle at another. We aim to develop methods to merge these data streams effectively, creating a unified understanding of the environment.

Additionally, a future interesting work is on mechanisms to verify obstacles. For example, an AIV could be assigned to physically check the presence of an obstacle, ensuring that the system has accurate and up-to-date information. This verification process will help address discrepancies and improve the reliability of obstacle detection.

To support these advancements, we plan to implement a shared memory system that enables AIV agents to access global information, such as task assignments and environmental maps. This system will allow for better coordination and task delegation, reducing the need for CRM (acknowledgment messages) and improving overall efficiency. Furthermore, the objective is to continue to develop our simulation platform (appendix B - B.4) to accommodate fleets of heterogeneous robots, each with different capabilities, ensuring that TA aligns with each robot's strengths.

By focusing on these approaches, we aim to significantly enhance the cooperation and operational autonomy of AIVs in real-world applications.

Cooperative Perception - Carla simulation

Our future work will focus on enhancing the integration of cooperative perception into AIV and industrial environments.

A primary goal is to advance our approach: generate V2X data with Carla simulator [VTC24] to feeding AI with relevant information by implementing a data clustering method that assigns confidence levels to inputs from various sensors, such as cameras. This method will help develop dynamic confidence metrics for sensor data, which is expected to support more adaptive and responsive multi-stakeholder systems.

We also plan to leverage Carla to simulate scenarios involving AIVs and infrastructure. By experimenting with different camera placements and evaluating their effects on scene analysis and decision-making quality, we aim to optimize deployment strategies and reduce

the time required to set up scenarios with active infrastructure. It will allow cooperation between robots and infrastructure on the model of cooperative ITS and reduce reliance on central servers.

To further evaluate this possibility, we will investigate how to integrate industrial vehicles into Carla. Moreover, with the ROS2 environment, we will be able to emulate simulated robots in the Carla environment, and for example work on co-simulation.

Additionally, we will explore how network conditions and data freshness influence data fusion processes. By using network simulators and message loss models, we intend to evaluate how these factors affect the integration and effectiveness of sensor data, which will contribute to improving the robustness and reliability of cooperative perception systems. Overall, our efforts are geared toward creating a more adaptive, efficient, and robust system for cooperative perception in industrial environments.

Energy management

We plan to continue integrating fuzzy models into our AIV simulation agents in order to increase the relevance and effectiveness of their decisions in the management of their energy recharge.

Moreover, another perspective would be to add learning capabilities (e.g., based on neural networks [Yud+19]) to them in order to increase the relevance and efficiency of their decisions in the collective management of their autonomies.

Integration of the different functionalities

A central question raised by my thesis is how to effectively integrate various functionalities to achieve global autonomy for AIVs. One promising direction for future research would involve testing different scenarios in a simulation environment, enabling AIVs agents to apply a range of collective strategies for achieving this goal.

One of the major perspectives of my work would be to test different scenarios in simulation with the possibility for AIV agents to be able to use all the different collective strategies to achieve overall autonomy. To do this, it will be necessary to first develop a complete multi-agent simulation platform with different possible circuits, and with the possibility of using the different collective strategies.

The broader objective of this research is to ultimately integrate these capabilities into robotic systems using ROS2 for example and perform experimentation with heterogeneous

fleets of robots. The integration of the capabilities to use collective strategies into physical robots would enable comprehensive testing of their effectiveness in real-world conditions.

Towards fleet-centered collective autonomy

Following the integration of different functionalities, a key direction for future research is to move from an individual AIV focused approach to a fleet-centered approach to autonomy. While in this manuscript we have focused more on individual AIVs who have benefited from collective strategies, the next step is to improve collaboration by focusing on the fleet as a whole, rather than on each AIV as a unit.

In this approach, fleet-level autonomy can be improved by running simulations that represent industrial scenarios where the fleet operates as a unified system. Instead of just tracking the performance of individual AIVs, we should create performance indicators that measure the fleet's overall efficiency. For instance, we could monitor the fleet's total energy usage, the number of packages handled per unit of time by the fleet, or how well the AIVs work together to share tasks and resources.

By treating the fleet as one coordinated system, we can significantly boost overall autonomy. Measuring and improving the fleet's ability to work as a cohesive unit will be key to achieving better performance in industrial settings. Shifting towards this fleet-centered approach in simulations and real-world tests is a crucial step in advancing collective autonomy for AIVs. Moreover, this perspective could enable us to refine our proposal scale of autonomy from a collective point of view for a fleet of AIVs (**subsection 1.3.3**).

Real field experiments

Finally, all contributions proposed are not validated with real field experiments, which is a big limitation. However, the "Alpha Project"¹ has emerged and ideas from my thesis work are being reused in this project. Its main objective is to improve the automation of baggage transport at airports using a fleet of AIVs. The main areas of research include system performance, operability, energy efficiency, safety and cybersecurity. Alstef Group seeks to enhance its solution by integrating skills, particularly in artificial intelligence, robot fleet management and cybersecurity. The work focuses in particular on developing algorithms for optimizing AIV routes and missions, inter-vehicle communication and intrusion detection to improve system security.

1. Alpha Project: <https://alstefgroup.com/fr/alpha-project/>

Additional key areas of focus include pedestrian management, building on the intersection simulation work using the Carla simulator (**subsection 4.3.1**), and energy management through the multi-agent simulator depicted in **appendix B - B.5**. Some methodological framework established during this thesis is therefore being further developed and applied within the Alpha Project.

SYMBOLS TABLE

Symbol	Description
$allMissionsCompleted$	The successful completion of all missions assigned to each agent. (see Equation (2.1)).
$completionTime_i$	The completion time of all assigned tasks for the robot i .
$distTravelled$	The total distance covered by each AIV agent.
κ	The task category (for example in a warehouse, move goods, i.e.: “load the goods at a starting point to bring it to an ending point”).
m_i	The mission i to be carried out.
n_i	The node i in an oriented graph.
s_i	The size of the obstacle.
Sc_i	The number of the scenario.
$nbMissions$	The number of missions carried out.
$nbRecharges$	The number of recharges performed.
p_{end}	The task ending point.
τ_{pr}	The task priority.
p_{start}	The task starting point.
$ratio$	The ratio of the number of tasks to be performed to the number of tasks fully completed by each AIV agent.
$robotTasksCompletionTime$	The makespan - the total time required for a robot to complete all assigned tasks (see Equation (1.1)).
σ	The task status.
τ	A task defined by the tuple described in Equation (3.1) .
τ_i	The task i .
τ_{id}	The task identifier.
$tasksCompleted$	Tasks fully completed by each AIV agent.

Symbol	Description
<i>totalCompletionTime</i>	The sum of costs - the total time expended by all robots to reach their respective tasks (see Equation (1.2)).
<i>tasksToBePerformed</i>	Tasks to be performed by each AIV agent.
<i>timeMission</i>	The average time taken to complete a mission in seconds.
<i>totalDistanceTravelled</i>	The cumulative distance travelled by all agents to complete the assigned missions (see Equation (2.2)).
t_{start}	The task starting time.
t_{end}	The task ending time.
<i>wtRecharging</i>	Waiting times for recharging in seconds.

Table A.1 – Symbols table

Linguistic Variable	Description
<i>Availability</i>	This variable represents the number of bags the AIV must handle, measured as a number from 0 to 20.
<i>AvailabilityStation1</i>	This variable represents the distance to station 1 the AIV must handle, measured as a number from 0 to 5 (corresponding to the number of AIVs in the simulation).
<i>AvailabilityStation2</i>	This variable represents the distance to station 2 the AIV must handle, measured as a number from 0 to 5 (corresponding to the number of AIVs in the simulation).
<i>Cost</i>	This variable ranges from 0 to 1 and determines whether the system should recharge or finish its current task.
<i>EnergyLevel</i>	This variable represents the energy level of the system, measured as a percentage between 0 and 100.
<i>DistanceToEnergySource</i>	This variable indicates the distance to the energy source in meters, ranging from 0 to 100.
<i>DistanceToStation1</i>	This variable represents the distance to station 1 that the AIV must consider, measured on a scale from 0 to 1200 (in meters).
<i>DistanceToStation2</i>	This variable represents the distance to station 2 that the AIV must consider, measured on a scale from 0 to 1200 (in meters).

Linguistic Variable	Description
<i>DistanceToTarget</i>	This variable measures the distance to the target in meters, ranging from 0 to 100.
<i>Decision</i>	This variable ranges from 0 to 1 and determines whether the system should recharge or finish its current task.
<i>ProximityAIV</i>	This variable represents the proximity of other AIV agents (use of observed and safety distances), measured as a number from 0 to 1000 (meters).
<i>Recharge</i>	This variable ranges from 0 to 1 and dictates whether the system should recharge or continue its current task.
<i>Speed</i>	This variable represents the speed of the AIV, measured in meters per second (m/s) within the range of 0 to 10.
<i>Urgency</i>	This variable represents the distance to the station (1 or 2) the AIV must handle, measured as a number from 0 to 10.

Table A.2 – Linguistic variables table

MULTI-AGENT SIMULATION PLATFORMS

B.1 "4-Lace Circuit" simulation

The interface of the "COSIVA Simulation" application is implemented in Python and designed to facilitate the simulation of AIVs navigating on situations that favour crossing traffic. Accounting for this characteristic led us naturally to diagram the circulation in four loops referred to as the "4-Lace Circuit" (shown in **Figure B.1**). The main simulation area displays the "4-Lace Circuit", characterized by four interconnected loops. Each loop intersects at designated points highlighted in red, indicating critical intersections.

Four differently colored dots (yellow, red, blue, and green) represent the AIV moving along the circuit paths. Below the simulation area, a console features buttons allowing users to select different predefined scenarios for the simulation. The button scenarios correspond to:

- *Scenario 0* (Sc_0): four AIV agents circulate continuously at the same speed without any need of cooperation for crossing intersections.
- *Scenario 1* (Sc_1): four AIV agents circulate continuously, independently (while exchanging messages to cross intersections) and at randomly changing speeds.
- *Scenario 2* (Sc_2): same context as Sc_1 where some obstacles are randomly generated on the circuit (spatial and temporal generation). The RADAR of an AIV agent can also perceive obstacles in the aisles ahead that constrain its path. On perceiving them, it cooperates to warn other AIVs by sending an `Obstacle_msg`, and then avoids the fixed obstacle if possible.

The *Obstacle* button, when pressed, introduces random obstacles of varying sizes (from s_1 to s_4) onto the circuit, adding a layer of complexity to the vehicle navigation. The "STOP" button halts the simulation, providing control over the ongoing process.

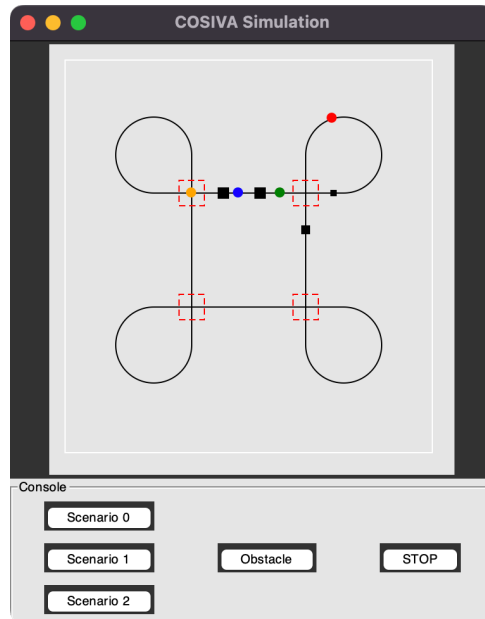


Figure B.1 – "4-Lace Circuit" simulation interface

B.2 ACT simulation

Our focus in this simulation was to check if the avoidance is well respected and the obstacles are perceived by the AIVs. We use this application simulation to validate the extended Bahnes's algorithm proposed in **chapter 2** with collision avoidance and fixed or dynamic obstacle detection processing.

We simulate our proposed algorithm staying within the framework of the three scenarios proposed by [BKH16].

The traffic plan presented in the interface application in **Figure B.2** and **Figure B.3** involves different intersections, where vehicles can arrive from different sides like in a warehouse (4 intersections are shown). Thus, it provides the different characteristics of an industrial environment and allows us to realize simulated experimental tests in line with realistic scenarios. It shows the different scenarios that we consider as a benchmark plan to compare results. Therefore, ten AIVs are distributed over 3 circuits: the red AIVs on the first circuit, the blue ones on the second and the yellow ones on the third.

We notice in the simulation that the avoidance is well respected and the obstacles are perceived by the AIVs. Therefore, this simulation validates the extended Bahnes's algorithm with collision avoidance and fixed or dynamic obstacle detection processing in

a more realistic circuit than in the "4-Lace Citcuit" application (**section B.1**).

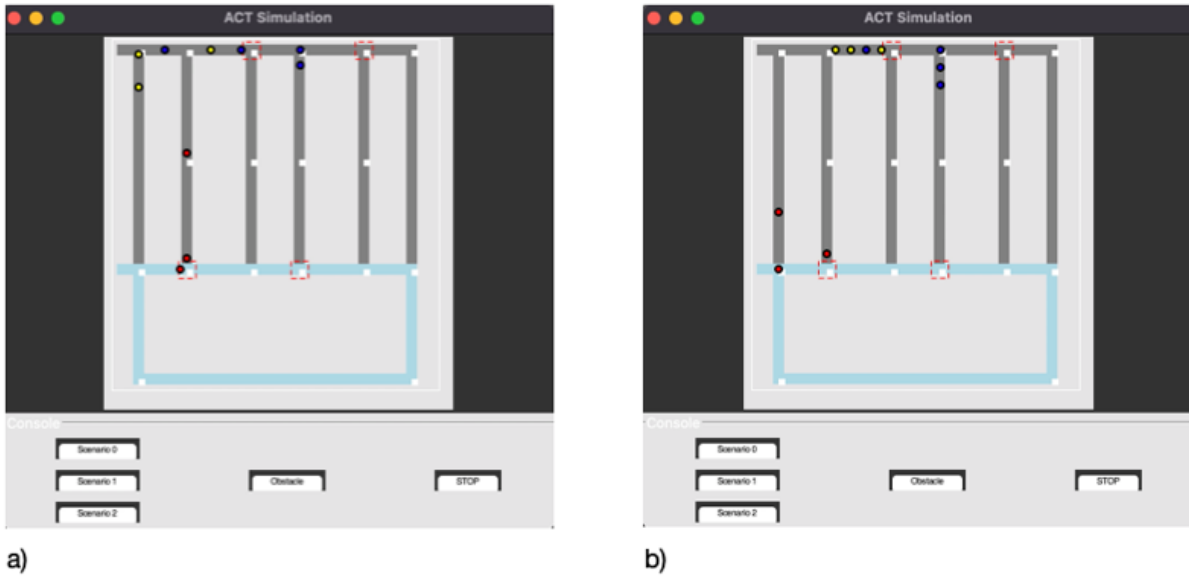


Figure B.2 – Simulation of radar use:

a - at the top of the picture: one blue and three yellow AIVs arrive near the intersection.
b - while waiting for the yellow AIV to pass the intersection, the radar of the blue AIV and the two other yellow AIV allow them to stop and keep their distance to avoid colliding.

B.3 Industrial circuit simulation

The "*IndustrialCircuit: Warehouse - Wholesaler*" application contains a specific circuit which includes several intersections, where vehicles can arrive from different sides like in a warehouse presented by [TBS18]. Thus, this kind of circulation plan provides the different characteristics of an industrial environment allowing us to conduct simulated experimental tests in accordance with realistic scenarios of an industrial context.

We implemented a Python graphical application with different functionalities. The different classes presented in **chapter 2, Figure 2.2** have been implemented in Python from scratch. Its interface is shown in **Figure B.4**. In the central part, we have the representation of the warehouse present in **chapter 2 - Figure 2.8**. The different nodes are represented by white squares in the interface. The vehicles are visualised by small colored circles and the obstacles are colored black with different sizes corresponding to the obstruction levels in the aisle. Five AIV agents are integrated into this environment corresponding to the five parking spaces available in this environment.

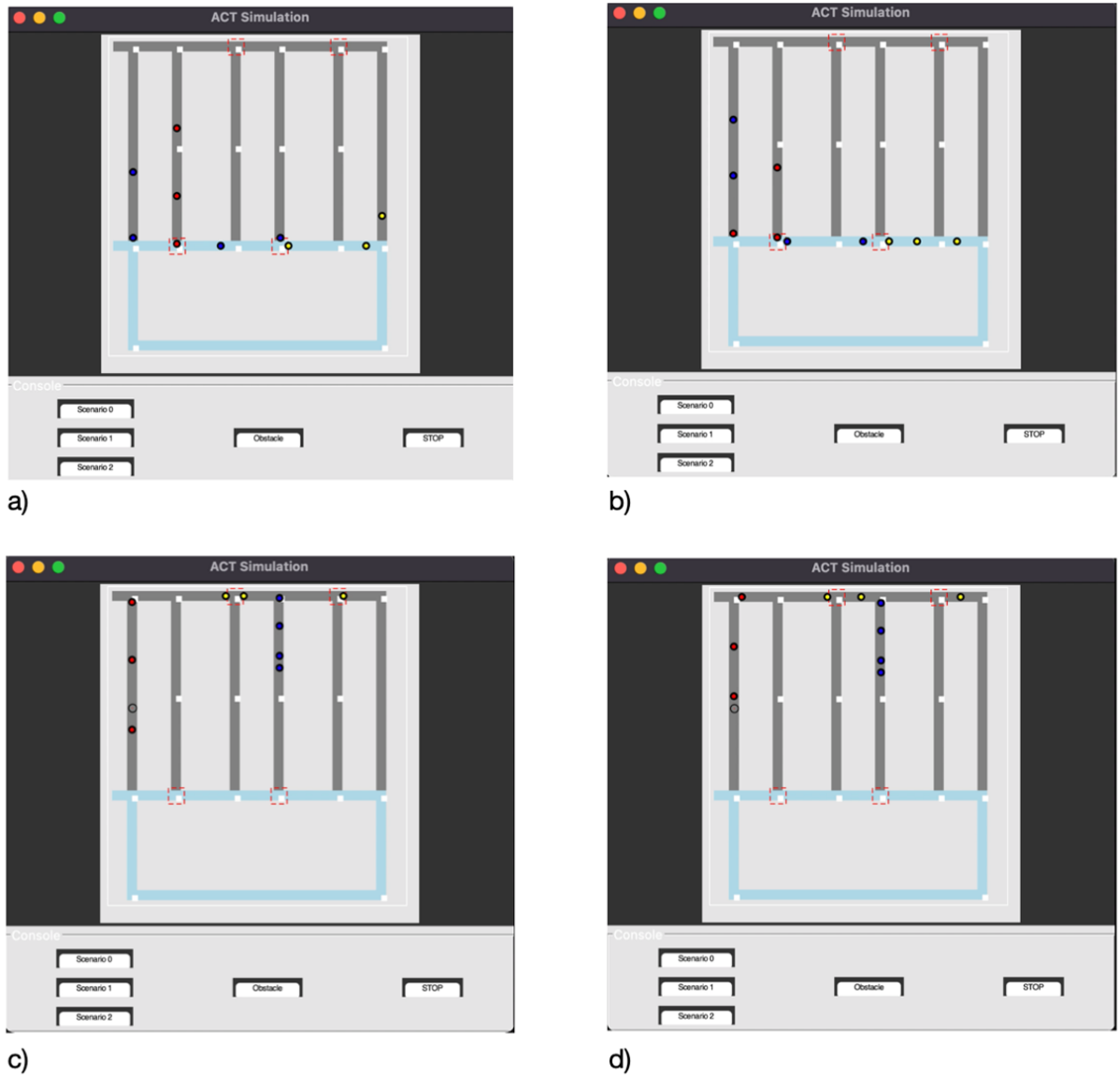


Figure B.3 – Simulation of the scenarios:

- a - in the center of the picture: a blue and yellow AIV arrive at an intersection.
- b - the yellow AIV passed the intersection after communicating with other AIVs.
- c - on the left side of the picture: a red AIV perceives a fixed obstacle in front of him.
- d - a red AIV avoided the obstacle

The functionalities proposed by our application (**Figure B.4**) are to:

- Stop the simulation.
- Place obstacles randomly in terms of size s_1 to s_4 (s_4 corresponding to a size that obstructs an entire aisle) and position at the six possible nodes shown in yellow.
- Launch a scenario: the agents must then carry out missions given by the supervisor. That is, go from their parking place to one of the three green nodes in **chapter 2 - Figure 2.8**, before returning to their parking place. On the other hand, pressing the *Scenario 0*, *1* or *2* button allows them to choose which strategy (Sc_1 , Sc_2 or Sc_3) they can perform. The description of each strategy is explained in **chapter 2 - section 2.3.2**.

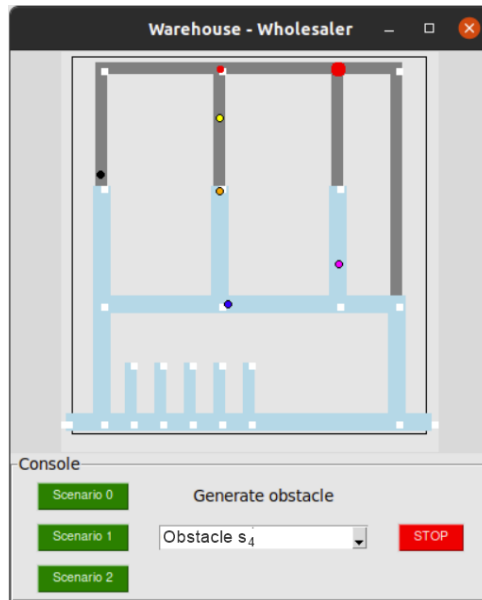


Figure B.4 – Application interface to simulate missions in the warehouse environment

B.4 Task allocation simulation

In order to simulate with a visualization of the task process of scenarios, we augmented the simulation interface presented in the above section, in **Figure B.4**. The augmented version developed is depicted in **Figure B.5**, and designed generically to integrate different types of traffic plans. Its implementation respects the agent model presented in **Figure 2.2** depicted in **chapter 2 - section 2.2.2**.

Additionally, this simulator is used for various laboratory experiments, for teaching engineering students and could be of interest to the community¹.

This interface is divided into five frames:

- **Frame 1:** visualization of the warehouse presented in figure **Figure 3.4**. The white squares represent tag agents, used by AIVs agents to locate themselves. AIV agents are visualized by colored circles, and obstacles are surrounded by a red circle of variable size corresponding to their level of obstruction in the aisle. The four camera agents of the infrastructure are identified by a black square evoking their viewing area of the aisle.
- **Frame 2:** application management and its various features. It is thus possible:
 1. to simulate the four types of scenarios illustrated in the following section, as well as a random scenario;
 2. to generate obstacles randomly on the circuit;
 3. to emulate a robot or camera failure;
 4. to view a model of the circuit with the node numbers, as in **Figure 3.5**, by clicking on the circuit button.
- **Frame 3:** supervision of AIV agents. This frame makes it possible to visualize the missions assigned to the various AIV agents, their paths, their statuses and other information useful to the supervisor agent. When an AIV agent plays the role of auctioneer, he is graphically identifiable by a frame.
- **Frame 4:** supervision of camera agents. This frame makes it possible to identify their status, their position and their detection of obstacles. When a camera agent has detected an obstacle, it is graphically identifiable by a black frame.
- **Frame 5:** task supervision. This frame makes it possible to monitor the progress of the performance of the tasks allocated to the AIVs. The states of the different task attributes are updated there: task identifiers, task starting point and ending point, and task states (attribute, in progress, blocked, completed).

The traffic plan chosen, and presented in the form of a directed graph in **Figure 3.5**, makes it possible to start a set of problem scenarios that can be easily configured in the interface.

1. In order to facilitate community utilization, the code of our project is available on the Gitlab page: https://gitlab.inria.fr/jgrosset/AIV_Simulator



Figure B.5 – Multi-agent simulation interface

B.5 Airport baggage handling simulation

The interface of our “Airport Baggage Handling Simulation” application, in **Figure B.6**, was developed to provide a comprehensive overview of the autonomous management and operation of AIVs for baggage handling. This intuitive and structured interface allows us to monitor and analyze the performance of AIVs in the circulation scenario.

The interface is divided into several sections, each displaying critical information about the simulation status and AIV performance. The different sections of the interface are as follows:

- **Energy Level Display:** Each AIV’s energy level is represented by a horizontal bar, which visually indicates the remaining battery life. For instance, *AIV3*’s bar is red, signifying a critical battery level, while the other AIVs have yellow bars, indicating varying levels of charge.
- **Charge Level Information:** Below the energy level display, this section provides detailed statistics on the recharging activities:
 - Recharge Time: The total time spent on recharging.

- Waiting Time: The total waiting time before recharging.
- Recharges per AIV: The number of recharge cycles completed by each AIV.
- **Simulation Area:** The central area of the interface depicts the simulation environment. It visually represents the positions and movements of the AIVs as they handle baggage within the airport layout. Different black points represent the location of each node corresponding to the oriented graph shown in **chapter 5 - Figure 5.7**.
- **Baggage Level Information:** This section displays baggage handling metrics:
 - Total Baggage: The total number of baggage that must be dealt with from the start of the scenario.
 - Waiting Baggage: The maximum number of baggage waiting to be processed as a moment.
 - Baggage per AIV: The number of baggage handled by each AIV.
- **Time Level Information:** Below the baggage level section, this displays time-related data:
 - Simulation Time: The total elapsed time of the simulation.
 - Missions per AIV: The average duration of missions completed by each AIV.

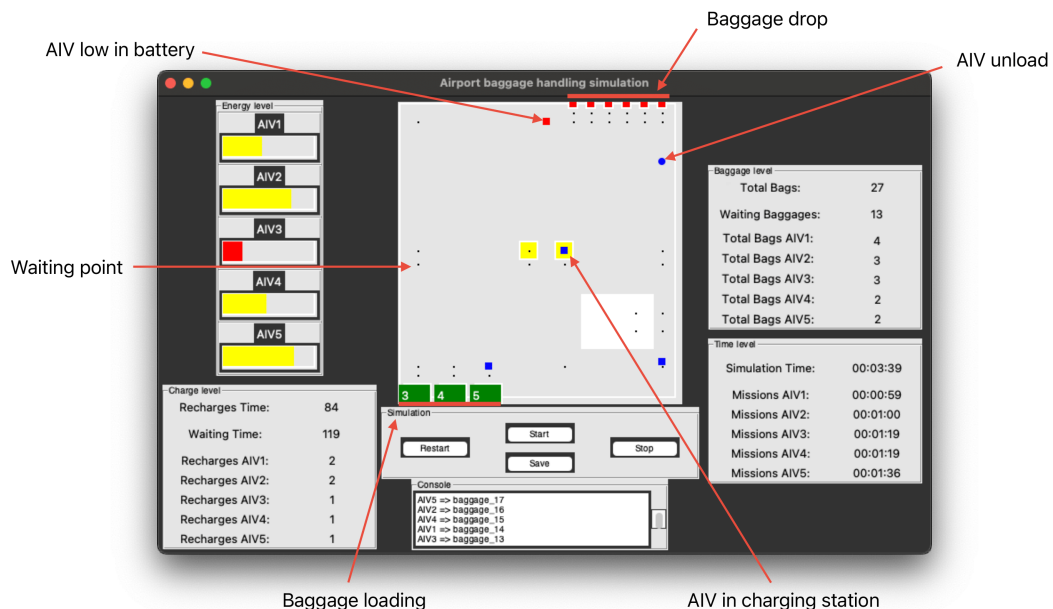


Figure B.6 – Airport baggage handling simulation HMI

This HMI application includes an intuitive “Experiment Parameters” window depicted in **Figure B.7**, providing users with essential options to set up and customize their simulation. This interface, which appears upon clicking the “Start” button, allows for comprehensive experiment configuration, enabling the precise control of various simulation parameters.

Experiment parameters:

- Save the Experiment Data: Users can opt to save the data generated during the experiment.
- Folder Name for Experiment Data: This input field allows users to specify the name of the folder where the experiment data will be saved.
- Number of Baggage: Users can define the number of baggage items to be handled by the AIVs during the simulation. The current value is set to 1000.
- Choice of Scenario: This dropdown menu provides a selection of scenarios from 1 to 8, as detailed in **chapter 5**. Users can choose the desired scenario to be simulated.
- Model Control for AIVs: This menu allows users to choose the control model for the AIVs. The selected model in the image is “Fuzzy_Basic_Model”, which likely represents a basic fuzzy logic model for AIV control.
- Cost of Calculation: Users can select the method for calculating the cost of the simulation for TA. The options include “Random,” “Normal,” and “Fuzzy Logic.” The “Normal” option is selected in the example.
- Start: Once all parameters are set, clicking this button will initiate the simulation based on the specified settings.

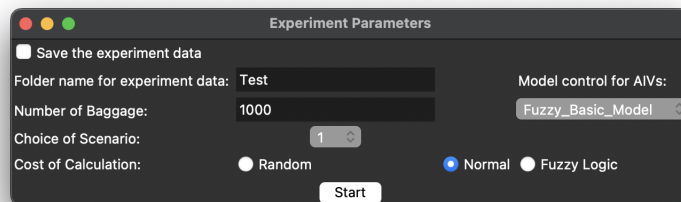


Figure B.7 – Airport baggage handling simulation parameters

METHOD FOR ESTIMATING AIV POSITIONS

C.1 Context

In the **chapter 2**, we proposed two critical contributions to the field: collective obstacle avoidance for AIVs within a fleet. Indeed, we proposed the augmented Bahnes' algorithm (**section 2.2**) for obstacle avoidance and a global collective avoidance strategy (**section 2.3**). Both of these strategies rely on the assumption that a vehicle can accurately determine its position within an environment. Effective obstacle avoidance and coordinated movement are predicated on precise localization to navigate safely and efficiently.

Moreover, the CAM messages standardized by ETSI are based on the strong assumption that a vehicle knows where it is, using GPS. However, in indoor environments, GPS does not work, and even more in disturbed industrial environments, GPS is not the tool used for AIVs to locate themselves. As we outlined in the **section 1.2.3**, vehicles must therefore be given other means to locate themselves. In our research, the measurements necessary for the estimation were susceptible to corruption by noise.

The result can be the generation of an input that introduces uncertainty into the inference. Uncertainty is, then, at the heart of the estimation problem: in the absence of uncertainty, many problems would have simple algebraic solutions [Cox89]. Before full-scale testing of traffic scenarios involving AIV or more complex traffic situations can begin, it is essential to consider the simulation involved.

In this appendix, we present an enhanced ABMS model for the localization of AIVs in **section C.2**. Then, in **section C.3**, we explained our computational approach which makes it possible to obtain the next position of the AIV on the circuit 4 – *LaceCircuit* given its current position. This appendix presents research currently under review, which has not yet been published [ALGO24].

C.2 Localization AIV model

Our focus lies on the organization aspects of the circulation area, AIV and localization in **Figure C.1**.

Indeed, this figure specifically highlights the different classes used for AIV localization, including *TrafficZone*, *Section*, *Position*, and *TimeStampedPosition*, in contrast to the model presented in **Figure 2.2**.

In **Figure C.1**, the *TrafficZone* model represents the overall environment in which the AIV operates. Within this traffic zone, the AIV navigates a designated pathway known as the *Circuit*. The *Circuit* defines the routes or pathways that AIVs follow and can vary in form depending on the specific application context.

The *TrafficZone* consists of two main components: the *Circuit* and a grid of *Beacons* arranged in a matrix formation. The spacing of the beacon grid is adjustable, allowing for customization to meet the specific needs of the environment and to enhance the precision of AIV localization.

Additionally, the agent aspect brings to the AIV its autonomous character which allows it, for example, to arm an internal timer to trigger various actions on its own initiative, such as transmitting its timestamped positional data. The AIV's location within the traffic zone is encapsulated in the *TimeStampedPosition* model, containing their geographical coordinates and timestamp information.

The simulation system aims to virtually reproduce the kinetics of an AIV on a circuit representation without making strong assumptions about its shape or profile. Hence, the circuit is conceptualized as a series of sections, each reflecting local topology. The section has a great importance in our modelling. AIV has a reference to the section of the circuit on which it is currently located. However, it is the current section of the AIV which will inform it about its next position. The initial circuit section is labelled *S0*.

Similarly, the *Section* model can be expanded to represent more specific types of *Section* models. Each *Section* model includes references to other sections on the circuit with which it is connected. Specifically, it maintains a reference to the section that precedes it, known as the *Previous Section (Previous)*, and to the section that follows it, referred to as the *Next Section (Next)*. Additionally, the *Section* model tracks the positions of its endpoints, identified as the *first end (Ex1)* and the *second end (Ex2)*.

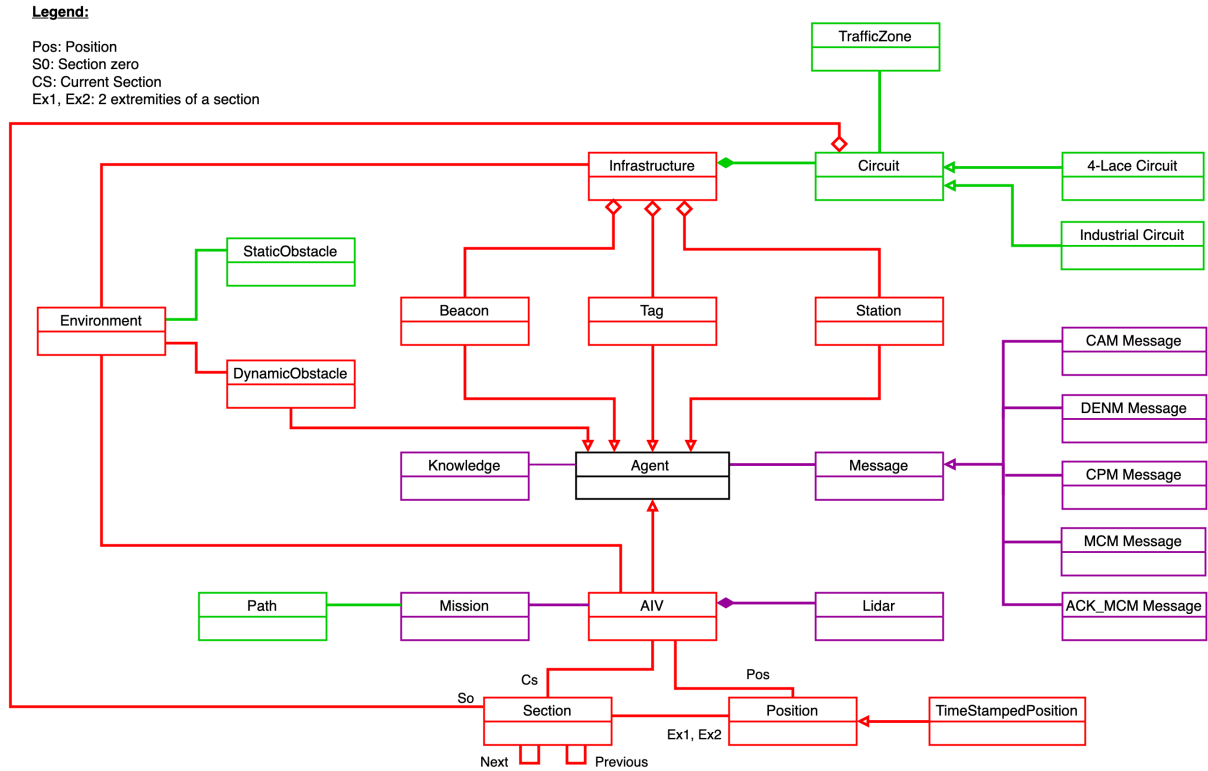


Figure C.1 – Agent-based model for AIVs obstacle avoidance and estimation positions

C.3 Mathematical position estimation model

Schematically, we defined the current position of the AIV as P_n and the next position as $P_n + 1$. Our approach was to define an abstract chunk model in which the updating of the current position would only be stated in principle, with no details were provided about the concrete implementation.

We then defined a concrete traffic section model by building on the aforementioned abstract model. The concrete section model served for the actual calculation of $P_n + 1$. Thus, we formally defined three types of concrete sections with their associated calculation intelligences:

- a circular arc section,
- a horizontal section,
- a vertical section.

Circular section

For a circular arc section with centre C and radius R , and for a time step Dt , the update of the position of the AIV is given by the **Equation (C.1)**.

$$\begin{cases} x_{n+1} = x_C + R \cos \left(\arctg \left(\frac{y_n - y_C}{x_n - x_C} \right) + \frac{v}{R} \Delta t \right) \\ y_{n+1} = y_C + R \sin \left(\arctg \left(\frac{y_n - y_C}{x_n - x_C} \right) + \frac{v}{R} \Delta t \right) \end{cases} \quad (\text{C.1})$$

Horizontal and vertical sections

Equations (C.2) and (C.3) give the required update for horizontal and vertical sections respectively.

$$\begin{cases} x_{n+1} = x_n + v \Delta t \\ y_{n+1} = y_n \end{cases} \quad (\text{C.2})$$

$$\begin{cases} x_{n+1} = x_n \\ y_{n+1} = y_n + v \Delta t \end{cases} \quad (\text{C.3})$$

AIV position update algorithm

The algorithm for updating the position of an AIV makes it possible to calculate the new position $P_{(n+1)}$ given its current position P_n . Before the update calculation, the Distance travelled on the *Current Section (DTCS)* is known. Here, the term “current section” means the section of the circuit on which the AIV is “currently” located. The *DTCS* can be updated by adding the *Distance To Travel (DTT)*, according to the expression $DTCS = DTCS + DTT$.

At this step, it is important to know if the current movement of the AIV remains or not within the limits of the current Section. To do this, we calculate the Length of the *Route Outside the Section (LROS)* according to the expression $LROS = DTCS - LCS$, where LCS is the Length of the Current Section. The *LROS* value can be negative or positive. If it is negative, then the next position $P_n + 1$ belongs to the current Section. Consequently, there is no *Section* change, and the AIV remains on the current Section on the circuit. It is then possible to calculate the new position $P_n + 1$, knowing the current one P_n .

It should be noted that the intelligence of the update of the position of the AIV is hosted in the model of the Section, which is normal because this calculation depends on

the topological profile of the considered Section.

Thus, the current section being referenced by the variable CS , and the position update function, being called "computeNextAIVPosition", the calculation of the next position of the AIV will be carried out, in a programming language such as python, thanks to the instruction $Pos = CS.computeNextAIVPosition(Pos, DTT)$, where the Pos variable designates the successive positions P_n and $P_n + 1$, and the incoming parameter DTT gives the function the *Distance To Travel*.

Conversely, if $LROS$ is positive, then the next position $P_n + 1$ belongs to the following Section. Therefore the AIV must continue on the next Section on the circuit. The next Section becomes the current Section, the first end ($Ex1$) of the current Section becomes the current *Position* of the AIV, the $DTCS$ is reset to zero, and the previously calculated $LROS$ becomes the new DTT . The position update algorithm, coded by the "computeNextPosition" function of class AIV, is presented by the flowchart of **Figure C.2**, in the case of a non-iterative modification of the variable DTT .

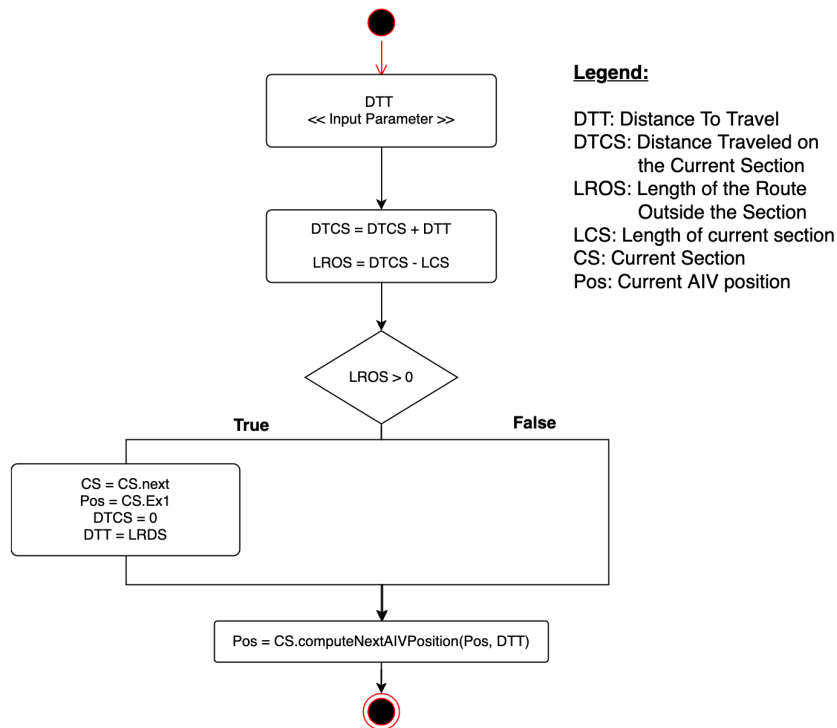


Figure C.2 – Algorithm for updating the position of an AIV agent

METHODOLOGY PROPOSED FOR ENERGY MANAGEMENT

D.1 Context

The variation in energy consumption is influenced by the acceleration and deceleration operations when using a stepping motor drive. As a general rule, the switching times for acceleration and deceleration have an impact on energy consumption. The choice of the cost function is crucial for optimizing problems in mobile robots. In some studies, the cost function is defined as the minimization of the energy supplied by the motor, represented by the integral of the squared motor force or the sum of the squared linear and angular velocities of the mobile robot. However, this function is optimal only for very small values, as it is more than a simple squared function. There are alternative cost functions for energy optimization, but they often include additional parameters, such as those related to the battery.

In this appendix, we are interested in a cost function that accommodates three types of motion simultaneously: acceleration, deceleration, and constant speed. The variation in the system's energy is defined by a force that represents these three motions. The change in force during the target phase also affects the work done. Work, in this context, refers to the means by which forces transfer energy. Therefore, the cost function is defined as the absolute work performed by the motor force.

This cost function enables us to tackle an optimization problem with a basic dynamic system. Despite the simplification of the mathematical model, a numerical method based on time discretization is not well-suited to the local environment of the application.

D.2 Methodology with a realistic point of view

This section proposes a methodology to address this problem, operating on varying of the maximum speed values under each node, using an optimal control problem. The biggest advantage of this methodology is that the numerical optimization methods are implemented to create the optimal maximum speed values. This technique reduces the execution time of this method in real-time.

The proposed methodology calculates the motor force, which serves as the command for the optimization problem. It also involves determining the moments of acceleration and deceleration. To define these moments, an algorithm is implemented to compute the acceleration and deceleration distances. This algorithm operates based on distance rather than time, making it suitable for implementation in the application.

Three strategies are generated by this methodology. With the help of two inputs: waiting baggage and traffic, a fuzzy logic model can select the most appropriate strategy for the robot during its mission.

D.2.1 Optimal control problem

This subsection addresses an Optimal Control Problem (OCP) for the optimization model, focusing on a mobile robot. The problem (OCP) combines a dynamic system, which is given by a mobile robot model, where the mobile robot is moving in a straight line and a cost function that is described by the absolute work of the motor force, which represents the amount of energy consumed by the robot during his mission.

Dynamic model of robot

The mobile robot model is derived from Newton's second law:

$$\sum F = M a$$

where F is the sum of all forces acting on an AIV, a is the linear acceleration of AIV and M is the total mass (AIV + baggage).

Based on the specific route of the AIVs, the system operates under the following simplifying assumptions:

- **Assumption 1:** An AIV moves in a straight line and does not consider curvilinear motion or slopes.

-
- **Assumption 2:** Air resistance is negligible.
 - **Assumption 3:** The energy consumed by the motor is included in the cost function, without constraints on the dynamic system.

Based on these assumptions, AIV dynamic system is described by a double integrator:

$$\begin{cases} \dot{x} = v \\ \dot{v} = \frac{F_T}{M} \end{cases} \quad (\text{D.1})$$

where, x , v , \ddot{x} are the longitudinal position, the linear velocity and the linear acceleration of AIV respectively. The ratio between force and mass $\frac{F_T}{M}$ represents the linear acceleration a . Then, the maximum force can be deduced from the high acceleration and the mass.

In the following, we assume that the forces acting on a AIV are composed of two forces:

$$F_T = F_m - F_r$$

where F_m is propelled by motor force and F_r is the resistance force.

Cost function

To account for the varying cost function due to the changes in the robot's movement (acceleration, deceleration, stopping) from the starting position to the final position over a period from T_o to final time T , we need to refine our model. Specifically, we will incorporate the power delivered by the robot's motor, which depends on the force.

The power P delivered by the robot's motor at any time t is given by:

$$P = F_m v \quad (\text{D.2})$$

The absolute work W done by the propulsion force over the period from T_0 to T :

$$W = \int_{T_o}^T |P(t)| dt = \int_{T_o}^T |F_m v| dt \quad (\text{D.3})$$

The OCP admits an optimal solution, and the linear speed v is always non-negative ($v \geq 0$) during the mission of the robot, as stated in [OB20].

Then, the cost function is given by:

$$\int_{T_o}^T |F_m| v dt \quad (\text{D.4})$$

Optimal control problems (OCP)

The formulated problem is defined by the cost function and the robot model presented above:

$$\left\{ \begin{array}{l} \text{Minimize} \quad W = \int_{T_o}^T |F_m| v dt \\ \text{Subject to} \\ \dot{x} = v \\ \dot{v} = \frac{F_m}{M} \\ x(T_o) = x_0, v(T_o) = v_0, v(T) = v_f, x(T) = x_f > 0 \\ T_o \text{ and } T : \text{fixed} > 0 \\ F_{min} \leq F_m \leq F_{max} \end{array} \right. \quad (\text{D.5})$$

Where x_0, v_0 : initial values of states, x_f, v_f : final values of states, F_{min} and F_{max} are the lower and upper bounds of the motor force F_m .

To achieve an optimal strategy for this control problem, various control techniques can be applied such as PMP [Kop62]. PMP is used to calculate the maximum speed values in the example of the optimal strategy is **Figure D.1** with its optimal speed profile, which was determined through the dynamic system. A solution is the *bang* type, if the control is equal to its maximum, or its minimum, which corresponds to the maximum acceleration or deceleration phase. An *inactivation* is defined when the control variable is null over a time interval, which corresponds to maintaining a constant speed.

Discrete time problem

The simulation of a realistic environment is carried out using digital data, which are represented in the form of sets of values such as vectors and matrices. To achieve high-performance simulations, a discrete model is best suited to closely mimic a real context. Therefore, we have discretised the problem (**Equation (D.5)**) [ALGO24]. Given N and define $h = \frac{T-T_o}{N}$, for $i = 0, \dots, N$.

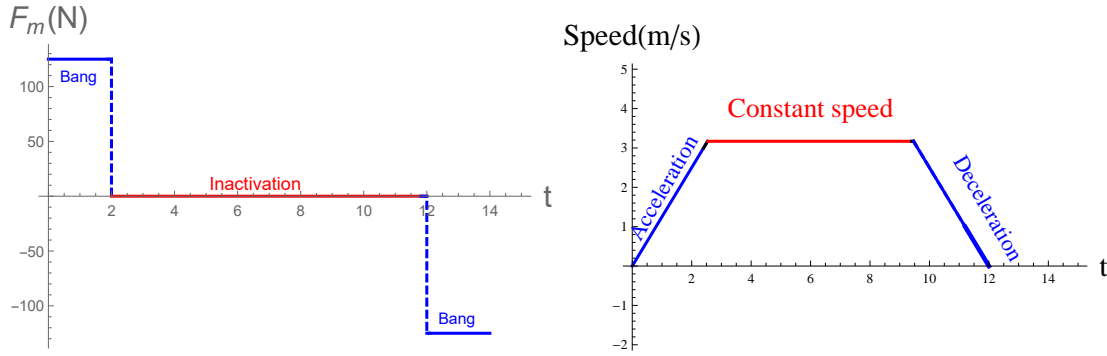


Figure D.1 – Example: optimal command for $T > T_{min}$ with $T_{min} = 9.79$ and speed over a distance 30 m

$$\left\{ \begin{array}{l} \text{Minimize} \quad W = \sum_{i=0}^N |F_m(i)| v(i) \\ \text{Subject to} \\ x(i+1) = x(i) + hv(i), \quad i = 0, \dots, N \\ v(i+1) = v(i) + h \frac{F_m(i)}{M}, \quad i = 0, \dots, N \\ x(0) = x_0, v(0) = v_0, v(N-1) = v_f, \quad x(N-1) = x_f > 0 \\ F_{min} \leq F_m(i) \leq F_{max}, \quad i = 0, \dots, N \end{array} \right. \quad (D.6)$$

D.2.2 Energy control strategies

Developing an energy control strategies algorithm based on an optimal speed profile that is defined from the optimal control problem. This algorithm is based on distance travelled, using basic principles of acceleration and deceleration, and applying these principles to small segments of the path. This method is practical for short distances between nodes and provides a systematic way to control the robot's movement efficiently. A more practical and simpler approach can be applied to determine the optimal control strategy for minimizing energy consumption.

By implementing three strategies based on the speed profile as a function of the robot's actual position. These speed profiles are generated by an algorithm capable of operating in real-time.

Proposed algorithm

We propose **Algorithm 3**, which connects two maximum speed values assigned to each node. This algorithm identifies two key distances—acceleration and deceleration distances—in order to link the two maximum speed values. The maximum speed values can be obtained using solutions generated by the PMP [Kop62]. However, these maximum speed values are optimal under certain constraints and need to be adapted to fit our specific application.

The following instructions outline the steps of the algorithm for comparing speeds, calculating distances, making decisions, and setting commands based on those decisions. Adjustments can be made according to specific requirements or additional constraints.

- Input:
 - Path nodes with associated maximum speed values.
 - Robot's initial position and speed.
- Output:
 - Real-time energy consumed and speed profile for the robot.
- 1. Initialization:
 - Identify the starting and ending nodes of the path.
 - Assign maximum speed values to each node using the PMP.
- 2. Calculate acceleration and deceleration distances:
 - For each pair of nodes, calculate the distance required for the robot to accelerate from the current speed to the maximum speed of the next node.
 - Similarly, calculate the distance required for deceleration.
- 3. Generate speed profile:
 - For each segment between nodes, generate a speed profile that smoothly transitions between the two maximum speed values.
 - Ensure the profile adheres to the calculated acceleration and deceleration distances.
- 4. Real-time adjustment:
 - Continuously monitor the robot's position and adjust the speed profile in real-time to account for any deviations or changes in the path.

In the application, step 1 "Offline Method: Table of Maximum Speed" represents the three strategies, which are included as a table or database of maximum speed for each node. The second step, "Online Method: Current Speed, Energy," will be called up in

Algorithm 3: Energy and Current Speed - Offline Method

Input: $M, h, a_{max}, \text{NextNode}, \text{CurrentDistance}$
Output: Energy, Current Speed

1 **Step 1** → **Offline Method: Table of Speed Max**
Input: M and $a_{max}, x_0, x_f, v_0, v_f$
Output: TableSpeedMax

2 **Function** [TableSpeedMax] = **Optimal-SpeedMax**(M and $a_{max}, x_0, x_f, v_0, v_f$)

3 $F_{max} = M * a_{max}$
4 $F_{min} = -M * a_{max}$
5 $T_{min} = f(M, F_{max}, F_{min}, x_0, x_f, v_0, v_f)$ //cf. equation (1)
6 Fixed $T > T_{min}$
7 $\Delta t = g(T, M, F_{max}, F_{min}, x_0, x_f, v_0, v_f)$ //cf. equation (2)
8 $SpeedMax = \frac{F_{max}}{M} \Delta t + v_0$

9 **end**

10 **Step 2** → **Online Method: CurrentSpeed, Energy**

11 **Function** Energy = **MotorEnergy**(Speed, FinalCommand)
12 Energy = Energy + abs(FinalCommand) * Speed
13 **end**

14 **Function** Speed = **CurrentSpeed**(Speed, FinalCommand, h, M)
15 Speed = Speed + $h * \frac{FinalCommand}{M}$

16 **end**

17 **Function** FinalCommand = **Command**(CurrentPosition, NextNode, Command_{sup}, Command_{inf}, Distance1, Distance2)

18 **if** (CurrentPosition – Distance1) ≤ 0 **then** FinalCommand = Command_{sup}
19 **else if** (NextNode – CurrentPosition – Distance2) ≤ 0 **then** FinalCommand = Command_{inf}
20
21 **else** FinalCommand = 0
22

23 **end**

24 **Function** [Command_{sup}, Command_{inf}, Distance1, Distance2] = **Acceleration-Deceleration**(PreviousSpeedMax, NextSpeedMax, NextNextSpeedMax, M, a_{max}, h)

25 $F_{max} = M * a_{max}$
26 $F_{min} = -M * a_{max}$

1. Compare the PreviousSpeedMax and NextSpeedMax values, NextSpeedMax and NextNextSpeedMax values to establish that this is an acceleration or deceleration decision.
2. Distance 1 is calculated from the first comparison and Distance 2 from the second one.
 - Distance1 = |PreviousSpeedMax – NextSpeedMax| * N1
 - Distance2 = |NextSpeedMax| * h
where N1 represents the number of points between PreviousSpeedMax and NextSpeedMax, depending on the step of discretization h .
3. Between each node, two decisions are defined, the command is given by: **if** Decision == Acceleration **then**
 - | Command_{sup} = F_{max}
 - else**
 - if** Decision == Deceleration **then** Command_{inf} = F_{min}
 - else** Constant Speed:
 - Command_{sup} = 0
 - Command_{inf} = 0

end

27 **end**

28 **Function** [PreviousSpeedMax, NextSpeedMax, NextNextSpeedMax] = **NodeSpeed**(TableSpeedMax, PreviousNode, NextNode, NextNextNode)

29 PreviousNode → PreviousSpeedMax
30 NextNode → NextSpeedMax
31 NextNextNode → NextNextSpeedMax

32 **end**

real-time under the fuzzy rule, which decides what type of strategy can be adapted to meet the constraints of waiting baggage and traffic flow. The energy constraint will be more advantageous for waiting baggage and traffic, which are defined as model inputs, with the three strategies as outputs of the fuzzy decision model in the next section. Finally, we define three strategies: *Strategy1*, *Strategy2*, and *Strategy3* which represent *LowEfficiency*, *MediumEfficiency*, and *StrongEfficiency*, respectively.

D.2.3 Fuzzy model for energy control

We propose a fuzzy logic model that allows us to determine which strategy to implement. The system uses two input variables and one output variable:

1. Input linguistic variable: **WaitingBaggage** - *Wb*
2. Input linguistic variable: **Traffic** - *Traffic*
3. Output linguistic variable: **Strategy** - *Strategy*

Input linguistic variables of the fuzzy decision model to determine the strategy

The fuzzy decision model relies on two primary input variables to make intelligent decisions regarding the strategy to use. These input variables are defined as follows:

- **Waiting Baggage** - this variable represents the waiting baggage level of the system, measured as a number of baggage from 0 to 1000. It is categorized into three linguistic terms:
 - *Low*: [0, 5]
 - *Medium*: [2, 5, 8]
 - *Strong*: [5, 1000]
- **Traffic** - this variable represents the traffic level of the system, measured as a number of AIVs from 0 to 5 (in our case study simulation). *Traffic*. It is categorized into three linguistic terms:
 - *Low*: [0, 2]
 - *Medium*: [1, 2, 3]
 - *Strong*: [2, 5]

Output linguistic variable of the fuzzy decision model to determine the strategy

The fuzzy decision model produces one output variable that guides the system's actions regarding energy management. This output variable enables the system to make informed decisions based on the input variables and the fuzzy rules, ensuring optimal performance and energy efficiency. The *Strategy* output variable ranges from 0 to 100 percentage of energy used and determines which strategy to follow. It is described using three linguistic terms:

- *LowEfficiency*: [0,20]
- *MediumEfficiency*: [0,30, 50]
- *StrongEfficiency*: [40, 100]

Fuzzy rules

The fuzzy rules establish the decision-making framework of the system. Each rule outlines conditions based on the input variables and determines the corresponding output decision. Our system adheres to the following rules:

1. If *Wb* is *Low* and *Traffic* is *Low* then *Strategy* is *LowEfficiency*.
2. If *Wb* is *Low* and *Traffic* is *Strong* then *Strategy* is *MediumEfficiency*.
3. If *Wb* is *Medium* and *Traffic* is *Medium* then *Strategy* is *MediumEfficiency*.
4. If *Wb* is *Medium* and *Traffic* is *Strong* then *Strategy* is *StrongEfficiency*.
5. If *Wb* is *Strong* and *Traffic* is *Low* then *Strategy* is *MediumEfficiency*.
6. If *Wb* is *Strong* and *Traffic* is *Strong* then *Strategy* is *StrongEfficiency*.

D.2.4 Numerical simulation in Matlab

From simulation data are adapted from the application environment, a step of discretization $h = 2$ as a minimum movement of the $5m$. In addition, it is suitable for airport environments, data provided by Alstef Group for Bagxone, for example: mass max is $100kg$, maximum speed is $7.5m/s$ and average speed is $5m/s$, maximum acceleration is $1.5m/s^2$ and maximum deceleration is $2m/s^2$. In the simulation for the three strategies, we defined the same values, except for the maximum acceleration and maximum deceleration are fixed at $1.25m/s^2$ for the application environment. To include the concept of minimum speed that is equal to $2.5m/s$.

Based on this knowledge, three strategies are generated thanks to these three speed values: minimum, average and maximum. Three strategies are developed to fit the different input linguistic variables of the fuzzy decision model, as noted in the previous section. *Strategy1* represents the first input "Waiting Baggage", when the level is low. When the level is strong and the second input "traffic" is low, it is *Strategy2* that satisfies these conditions. *Strategy3* describes a traffic strong. **Figure D.2** shows the path travelled for each of the strategies.

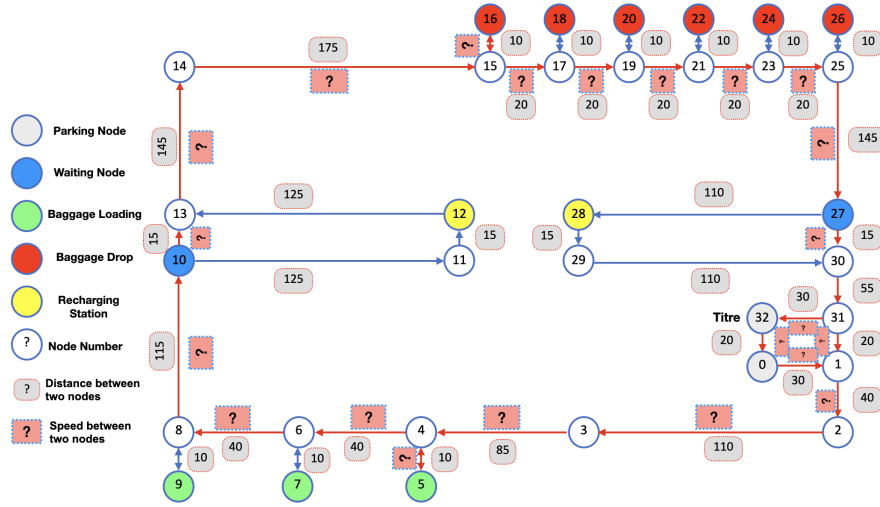


Figure D.2 – Map of Strategies in red color: oriented graph nodes

Strategy 1

We set Dis representing the distance between two successive nodes. From this distance, maximum speed values are defined for *Strategy1*. **Figure D.1** shows how we can obtain the value of the maximum speed, which is $2.5m/s$, for the path from *Node0* to *Node1* with a distance of $30m$ (see **Figure D.5**). *Strategy1* is defined according to these speed values in the **Equation (D.7)**.

$$Speed - Strategy1 = \begin{cases} 5m/s & \text{if } Dis \leq 100m \\ 2.5m/s & \text{Otherwise} \end{cases} \quad (D.7)$$

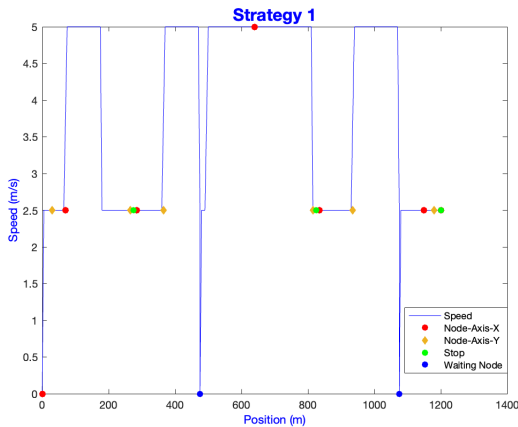


Figure D.3 – Position-Speed phase diagram: *Strategy1*

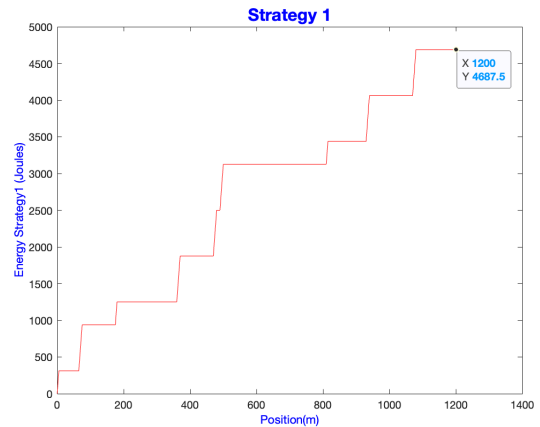


Figure D.4 – Energy consumed by the *Strategy1*

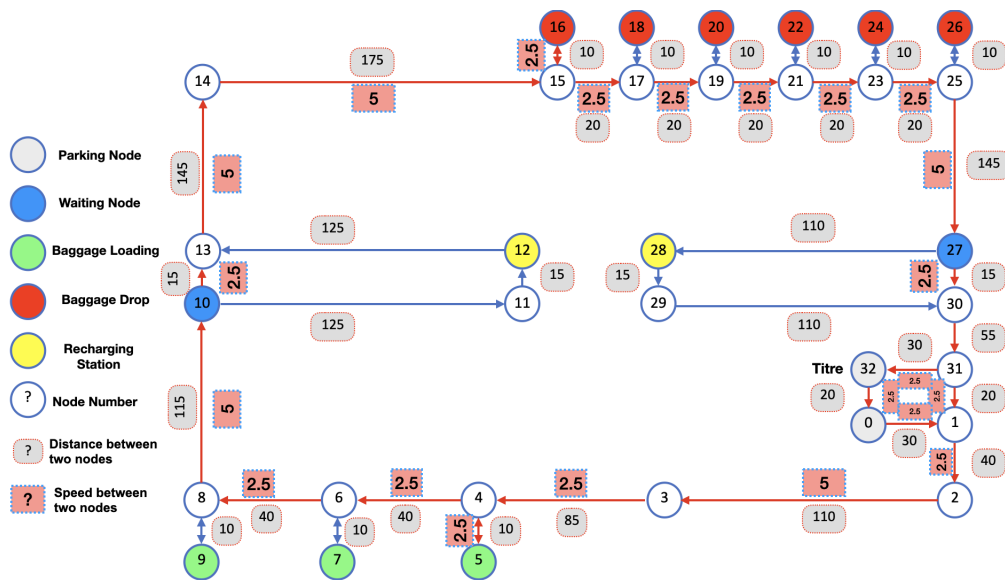


Figure D.5 – Map *Strategy1*: oriented graph nodes with their values distance (grey color) and speed (red color)

Strategy 2

Unlike *Strategy1*, where speed values are fixed in relation to distance between two nodes, *Strategy2* is arbitrarily defined with its maximum speed achieved. In *Strategy2*, high speed is the main speed used throughout the entire circuit. On the other hand, in *Strategy1*, the speed is limited to the average speed. The corresponding equation for *Strategy2* is shown in **Equation (D.8)**.

$$Speed - Strategy2 = \begin{cases} 7.5m/s & \text{if } Dis \leq 100m \\ 5m/s & \text{if } 50 \leq Dis < 100m \\ 2.5m/s & \text{Otherwise} \end{cases} \quad (D.8)$$

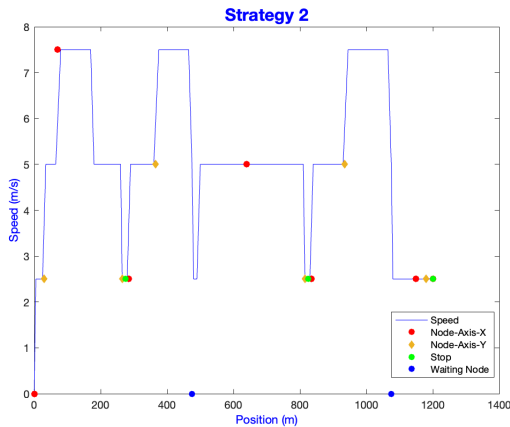


Figure D.6 – Position-Speed phase diagram: *Strategy2*

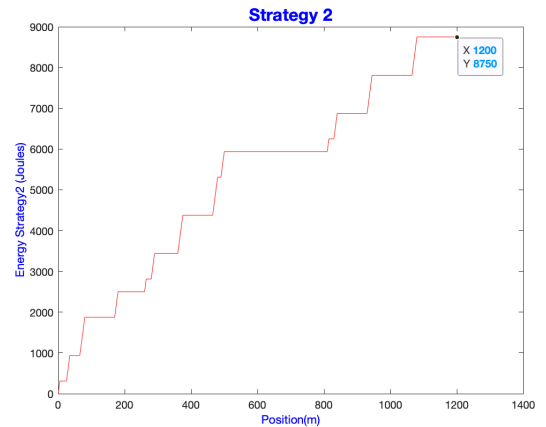


Figure D.7 – Energy consumed by the *Strategy2*

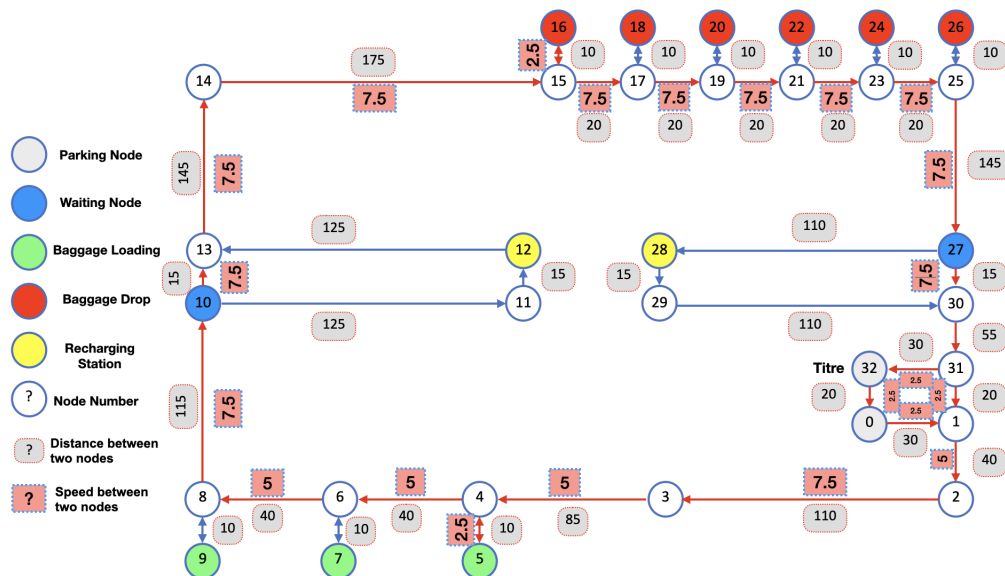


Figure D.8 – Map *Strategy2*: oriented graph nodes with their values distance (gray color) and speed (red color)

Strategy 3

The *Strategy3* is approximately a mix of *Strategy1* and *Strategy2*, described in **Equation (D.9)**.

$$Speed - Strategy3 = \begin{cases} 7.5m/s & \text{if } Dis \leq 100m \\ 5m/s & \text{if } 50 \leq Dis < 100m \\ 2.5m/s & \text{Otherwise} \end{cases} \quad (D.9)$$

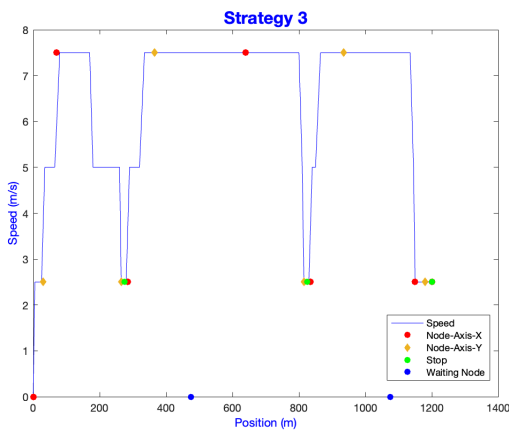


Figure D.9 – Position-Speed phase diagram: *Strategy3*

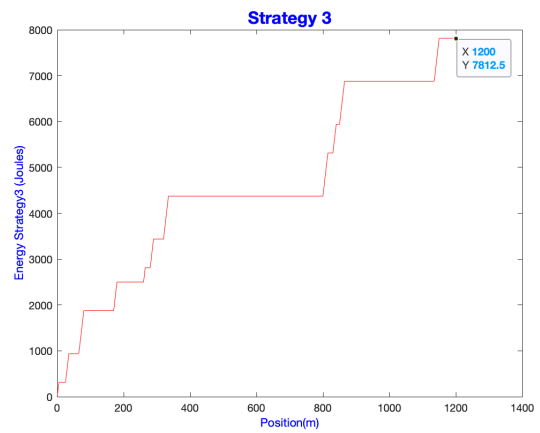


Figure D.10 – Energy consumed by the *Strategy3*

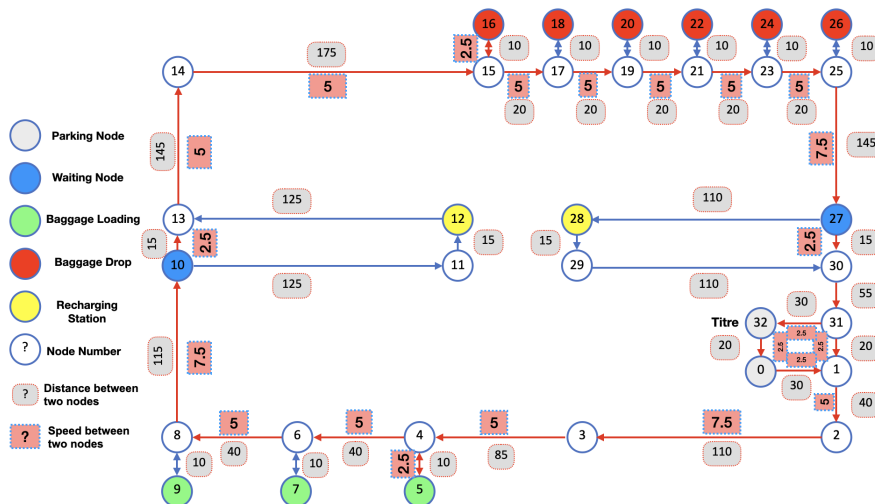


Figure D.11 – Map *Strategy3*: oriented graph nodes with their values distance (gray color) and speed (red color)

FUZZY AGENT-BASED SIMULATION

An agent-based system is fuzzy if its agents have fuzzy behaviors or if the knowledge they use is fuzzy [OFF12]. This means that agents can have:

1. fuzzy knowledge (fuzzy decision rules, fuzzy linguistic variables, and fuzzy linguistic values);
2. fuzzy behaviors (the behaviors adopted by agents because of fuzzy inferences);
3. fuzzy interactions, organizations, or roles.

The different elements of the fuzzy agent model are as follows [Fou13]:

- the agent-based fuzzy system (**section E.1**);
- the behaviour of a fuzzy agent, inspired by perceive-decide-act feedback loops [Bru+09] (**section E.2**);
- the behavioural functions of a fuzzy agent (**section E.3**);
- and the fuzzy interactions between two fuzzy agents (**section E.4**).

E.1 Agent-based fuzzy system

$$\tilde{M}_\alpha = \langle \tilde{A}, \tilde{I}, \tilde{P}, \tilde{O} \rangle (1) \quad (\text{E.1})$$

In **Equation (E.1)**: A is a set of agents, $A = \{\alpha_1, \dots, \alpha_n\}$; \tilde{A} is a set of fuzzy agents, $\tilde{A} = \{\tilde{\alpha}_1, \dots, \tilde{\alpha}_m\}$ with $\tilde{A} \subseteq A$; \tilde{I} is a set of fuzzy interactions between fuzzy agents; \tilde{P} is a set of fuzzy roles filled by fuzzy agents; and \tilde{O} is a set of fuzzy organisations defined for fuzzy agents (subsets of strongly related fuzzy agents).

E.2 Behaviour of a fuzzy agent

$$\tilde{\alpha}_i = \langle \Phi_{\Pi(\tilde{\alpha}_i)}, \Phi_{\Delta(\tilde{\alpha}_i)}, \Phi_{\Gamma(\tilde{\alpha}_i)}, K_{\tilde{\alpha}_i} \rangle \quad (\text{E.2})$$

In **Equation (E.2)**: for a fuzzy agent $\tilde{\alpha}_i$, $\langle \Phi_{\Pi(\tilde{\alpha}_i)}$ is its observation function, $\Phi_{\Delta(\tilde{\alpha}_i)}$ is its decision-making function, $\Phi_{\Gamma(\tilde{\alpha}_i)}$ is its action function, and $K_{\tilde{\alpha}_i}$ is its knowledge base.

E.3 Behavioural functions of a fuzzy agent

$$\Phi_{I\tilde{\alpha}_i} : (E_{\tilde{\alpha}_i} \cup I_{\tilde{\alpha}_i}) \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Pi_{\tilde{\alpha}_i} \quad (\text{E.3})$$

$$\Phi_{\Delta\tilde{\alpha}_i} : \Pi_{\tilde{\alpha}_i} \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Delta_{\tilde{\alpha}_i} \quad (\text{E.4})$$

$$\Phi_{\Gamma(\tilde{\alpha}_i)} : \Delta_{\tilde{\alpha}_i} \times \Sigma \rightarrow \Gamma_{\tilde{\alpha}_i} \quad (\text{E.5})$$

Equations (E.3), (E.4), and (E.5) define key functions for a fuzzy agent $\tilde{\alpha}_i$ within a fuzzy multi-agent system $\tilde{M}\alpha$. Specifically:

- **Equation (E.3)** represents the observation function, which maps the set of observed fuzzy events $E_{\tilde{\alpha}_i}$ and interactions $I_{\tilde{\alpha}_i}$, along with the agent's state $\Sigma_{\tilde{\alpha}_i}$, to its fuzzy perceptions $\Pi_{\tilde{\alpha}_i}$.
- **Equation (E.4)** defines the decision-making function, mapping fuzzy perceptions $\Pi_{\tilde{\alpha}_i}$ and agent state $\Sigma_{\tilde{\alpha}_i}$ to fuzzy decisions $\Delta_{\tilde{\alpha}_i}$.
- **Equation (E.5)** describes the action function, which maps fuzzy decisions $\Delta_{\tilde{\alpha}_i}$ and the overall system state Σ to the agent's fuzzy actions $\Gamma_{\tilde{\alpha}_i}$.

Here, $E_{\tilde{\alpha}_i}$ denotes the set of fuzzy events observed by the agent, $I_{\tilde{\alpha}_i}$ represents its fuzzy interactions, $\Sigma_{\tilde{\alpha}_i}$ is the agent's fuzzy state space, $\Pi_{\tilde{\alpha}_i}$ represents the set of fuzzy perceptions, $\Delta_{\tilde{\alpha}_i}$ denotes the fuzzy decisions made by the agent, $\Gamma_{\tilde{\alpha}_i}$ represents the agent's fuzzy actions, and Σ refers to the overall state of the fuzzy multi-agent system.

E.4 Fuzzy interactions between two fuzzy agents

$$\tilde{l}_l = \langle \tilde{\alpha}_s, \tilde{\alpha}_r, \tilde{\gamma}_c \rangle \quad (\text{E.6})$$

In **Equation (E.6)**: for fuzzy interaction \tilde{l}_l , $\tilde{\alpha}_s$ is the fuzzy source agent, $\tilde{\alpha}_r$ is the destination fuzzy agent, and $\tilde{\gamma}_c$ is a fuzzy communication act (for instance: inform, diffuse, ask, reply).

BIBLIOGRAPHY

- [AA21] Noor Abdul Khaleq and Ahmed Al-Araji, « A One Decade Survey of Autonomous Mobile Robot Systems », *in: International Journal of Electrical and Computer Engineering* 11 (Dec. 2021), pp. 4891–4906, DOI: 10.11591/ijece.v11i6.pp4891-4906.
- [AB21] Mohammed Alshaboti and Uthman Baroudi, « Multi-Robot Task Allocation System: Fuzzy Auction-Based and Adaptive Multi-threshold Approaches », *in: SN Computer Science* 2.2 (Feb. 2021), p. 87, ISSN: 2661-8907, DOI: 10.1007/s42979-021-00479-x.
- [AISYS24] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Fuzzy Agent-Based Simulation of Integrated Solutions for Task Allocation and Battery Charge Management for Fleets of Autonomous Industrial Vehicles », *in: AI-based Systems and Services (AISyS) 2024*, Venice, Italy, Oct. 2024.
- [ALGO24] Juliette Grosset, Ouzna Oukacha, *et al.*, « Fuzzy Multi-Agent Simulation for Collective Energy Management of Autonomous Industrial Vehicle Fleets », *in: Algorithms* 17.11 (Nov. 2024), p. 484, ISSN: 1999-4893, DOI: 10.3390/a17110484.
- [AM13] Wilmer Arellano and Imad Mahgoub, « TrafficModeler Extensions: A Case for Rapid VANET Simulation Using, OMNET, SUMO, and VEINS », *in: 2013 High Capacity Optical Networks and Emerging/Enabling Technologies*, Magosa, Cyprus: IEEE, Dec. 2013, pp. 109–115, ISBN: 978-1-4799-2569-8 978-1-4799-2568-1, DOI: 10.1109/HONET.2013.6729767.
- [And+15] Henrik Andreasson *et al.*, « Autonomous Transport Vehicles: Where We Are and What Is Missing », *in: IEEE Robotics and Automation Magazine* 22.1 (Mar. 2015), pp. 34–75, ISSN: 1070-9932, DOI: 10.1109/MRA.2014.2381357.
- [AOH20] Marwan Alakhras, Mourad Oussalah, and Mousa Hussein, « A Survey of Fuzzy Logic in Wireless Localization », *in: EURASIP Journal on Wireless Communications and Networking* 2020.1 (May 2020), p. 89, ISSN: 1687-1499, DOI: 10.1186/s13638-020-01703-7.

-
- [ARAC24] J. Grosset, A.-J. Fougères, C. Couturier, *et al.*, « Collision and Obstacle Avoidance for Industrial Autonomous Vehicles – Simulation and Experimentation Based on a Cooperative Approach. », *in: Advances in Robotics and Automatic Control*, Sergey Y. Yurish, International Frequency Sensor Association (IFSA) Publishing, vol. 3, 2024, pp. 1–23, ISBN: 978-84-09-57872-6.
- [ARCI22] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, Christophe Couturier, *et al.*, « Simulation of a Collision and Obstacle Avoidance Algorithm for Cooperative Industrial Autonomous Vehicles », *in: 2nd IFSA Winter Conference on Automation, Robotics & Communications for Industry 4.0 (ARCI' 2022)*, Andorra la Vella, Andorra, Feb. 2022.
- [ASPAI22] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Collective Obstacle Avoidance Strategy - an Agent-Based Simulation Approach », *in: ASPAI 2022: 4th International Conference on Advances in Signal Processing and Artificial Intelligence*, Corfu, Greece, Oct. 2022.
- [ASPAI24] Juliette Grosset, Alain-Jérôme Fougères, Moïse Djoko-Kouam, and Jean-Marie Bonnin, « Fuzzy Agent-Based Simulation for Managing Battery Recharging for a Fleet of Autonomous Industrial Vehicles », *in: ASPAI 2024: 6th International Conference on Advances in Signal Processing and Artificial Intelligence*, Funchal (Madeira), Portugal, Apr. 2024, **Best Paper Award**.
- [Ayn+17] Claude Aynaud *et al.*, « Real-Time Multisensor Vehicle Localization: A Geographical Information System Based Approach », *in: IEEE Robotics and Automation Magazine* 24.3 (Sept. 2017), pp. 65–74, DOI: 10.1109/MRA.2017.2669399.
- [Bar+03] Humberto Barberá *et al.*, *I-Fork: A Flexible AGV System Using Topological and Grid Maps*, vol. 2, Oct. 2003, 2152 vol.2, ISBN: 978-0-7803-7736-3, DOI: 10.1109/ROBOT.2003.1241911.
- [Bau+01] Bernhard Bauer *et al.*, « Agent UML: A Formalism for Specifying Multiagent Interaction », *in: Springer, Berlin 1957* (2001), pp. 91–103.
- [Baz+19] Alessandro Bazzi *et al.*, « Survey and Perspectives of Vehicular Wi-Fi versus Sidelink Cellular-V2X in the 5G Era », *in: Future Internet* 11.6 (June 2019), p. 122, ISSN: 1999-5903, DOI: 10.3390/fi11060122.

-
- [BCG07] Fabio Bellifemine, Giovanni Caire, and Dominic Greenwood, « Developing Multi-agent Systems with JADE », *in: Developing Multi-Agent Systems with JADE* (Feb. 2007), pp. 1–286, ISSN: 9780470057476, DOI: 10.1002/9780470058411.
- [BKD18] Rabab Benotsmame, György Kovács, and László Dudás, « The Concept of Autonomous Systems in Industry 4.0 », *in: Advanced Logistic Systems - Theory and Practice 12.1* (2018), pp. 77–87, ISSN: 1789-2198, DOI: 10.32971/als.2019.006.
- [BKD19] Rabab Benotsmame, György Kovács, and László Dudás, « Economic, Social Impacts and Operation of Smart Factories in Industry 4.0 Focusing on Simulation and Artificial Intelligence of Collaborating Robots », *in: Social Sciences 8* (May 2019), p. 143, DOI: 10.3390/socsci8050143.
- [BKH16] Nacera Bahnes, Bouabdellah Kechar, and Hafid Haffaf, « Cooperation between Intelligent Autonomous Vehicles to Enhance Container Terminal Operations », *in: Journal of Innovation in Digital Ecosystems*, Special Issue on Pattern Analysis and Intelligent Systems – With Revised Selected Papers of the PAIS Conference 3.1 (June 2016), pp. 22–29, ISSN: 2352-6645, DOI: 10.1016/j.jides.2016.05.002.
- [BM16] Roger Bostelman and Elena Messina, « Towards Development of an Automated Guided Vehicle Intelligence Level Performance Standard », *in: Apr. 2016*, pp. 1–22, ISBN: 978-0-8031-7633-1, DOI: 10.1520/STP159420150054.
- [Bru+09] Yuriy Brun *et al.*, « Engineering Self-Adaptive Systems through Feedback Loops », *in: Software Engineering for Self-Adaptive Systems*, ed. by Betty H. C. Cheng *et al.*, Berlin, Heidelberg: Springer, 2009, pp. 48–70, ISBN: 978-3-642-02161-9, DOI: 10.1007/978-3-642-02161-9_3.
- [C-R16] C-Roads Platform, "C-Roads Platform Final Report", tech. rep., Jan. 2016.
- [Cae+20] Holger Caesar *et al.*, « nuScenes: A Multimodal Dataset for Autonomous Driving », *in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11621–11631.
- [CAF17] Yang Chen, Arturo Ardila-Gomez, and Gladys Frame, « Achieving Energy Savings by Intelligent Transportation Systems Investments in the Context of Smart Cities », *in: Transportation Research Part D: Transport and Environment 54* (July 2017), pp. 381–396, ISSN: 1361-9209, DOI: 10.1016/j.trd.2017.06.008.

-
- [Cam+20] Sean Campbell *et al.*, « Where Am I? Localization Techniques for Mobile Robots A Review », *in: 2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, Barcelona, Spain: IEEE, Feb. 2020, pp. 43–47, ISBN: 978-1-72815-739-9, DOI: 10.1109/ICMRE49073.2020.9065135.
- [CAR] CAR-2-CAR, *CAR 2 CAR Communication Consortium*, <https://www.car-2-car.org/about-c-its>.
- [Car+17] Elena Cardarelli *et al.*, « Cooperative Cloud Robotics Architecture for the Coordination of Multi-AGV Systems in Industrial Warehouses », *in: Mechatronics* 45 (Aug. 2017), pp. 1–13, ISSN: 0957-4158, DOI: 10.1016/j.mechatronics.2017.04.005.
- [CCW19] Con Cronin, Andrew Conway, and Joseph Walsh, « State-of-the-Art Review of Autonomous Intelligent Vehicles (AIV) Technologies for the Automotive and Manufacturing Industry », *in: 2019 30th Irish Signals and Systems Conference (ISSC)*, June 2019, pp. 1–6, DOI: 10.1109/ISSC.2019.8904920.
- [CFK96] Y. Uny Cao, Alex S. Fukunaga, and Andrew B Kahng, « Cooperative Mobile Robotics: Antecedents and Directions », *in: Autonomous Robots* (1996), p. 21.
- [CG15] Gareth Cawood and Igor Gorlach, *Navigation and Locomotion of a Low-Cost Automated Guided Cart*, Nov. 2015, p. 88, DOI: 10.1109/RoboMech.2015.7359503.
- [Che+21] Che-Wen Chen, Chun-Liang Lin, *et al.*, « Design and Implementation of AMR Robot Based on RGBD, VSLAM and SLAM », *in: 2021 9th International Conference on Orange Technology (ICOT)*, Dec. 2021, pp. 1–5, DOI: 10.1109/ICOT54518.2021.9680621.
- [Che+22] Yiyang Chen, Chuanxin Cheng, *et al.*, « A Neural Network-Based Navigation Approach for Autonomous Mobile Robot Systems », *in: Applied Sciences* 12.15 (Jan. 2022), p. 7796, ISSN: 2076-3417, DOI: 10.3390/app12157796.
- [Cor+21] Alejandro Correa *et al.*, « On the Impact of V2X-based Maneuver Coordination on the Traffic », *in: 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)* (2021), DOI: 10.1109/VTC2021-Spring51267.2021.9448700.
- [Cox89] Ingemar J. Cox, « Blanche: Position Estimation For An Autonomous Robot Vehicle », *in: Proceedings. IEEE/RSJ International Workshop on Intelligent Robots and Systems ' (IROS '89) 'The Autonomous Mobile Robots and Its Applications*, Sept. 1989, pp. 432–439, DOI: 10.1109/IROS.1989.637941.

-
- [Dao+21] Alaa Daoud *et al.*, « ORNInA: A Decentralized, Auction-Based Multi-Agent Coordination in ODT Systems: Online Rescheduling with Neighborhood Exchange, Based on Insertion Heuristic and Auctions », *in: AI Communications* 34 (Jan. 2021), pp. 1–17, DOI: 10.3233/AIC-201579.
- [dDD15] Fernando de Lucca Siqueira, Patricia Della Mea Plentz, and Edson Roberto De Pieri, « A Fuzzy Approach to the Autonomous Recharging Problem for Mobile Robots », *in: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Changsha, China, 13/2016-08-15, pp. 1065–1070, DOI: 10.1109/FSKD.2016.7603326.
- [DF23] Moïse Djoko-Kouam and Alain-Jérôme Fougères, « Towards a Bijective Co-simulation Model Between Physical and Virtual Environments, Adapted to a Platform for Autonomous Industrial Vehicles », *in: Automation, Control and Intelligent Systems* 11.2 (July 2023), pp. 27–44, DOI: 10.11648/j.acis.20231102.12.
- [Dia+06] M. Bernardine Dias *et al.*, « Market-Based Multirobot Coordination: A Survey and Analysis », *in: Proceedings of the IEEE* 94 (Aug. 2006), pp. 1257–1270, DOI: 10.1109/JPROC.2006.876939.
- [DKJ18] Ali Dorri, Salil S. Kanhere, and Raja Jurdak, « Multi-Agent Systems: A Survey », *in: IEEE Access* 6 (2018), pp. 28573–28593, ISSN: 2169-3536, DOI: 10.1109/ACCESS.2018.2831228.
- [Dos+17] Alexey Dosovitskiy *et al.*, « CARLA: An Open Urban Driving Simulator », *in: Proceedings of the 1st Annual Conference on Robot Learning*, PMLR, Oct. 2017, pp. 1–16.
- [Dra+16] Ivica Draganjac *et al.*, « Decentralized Control of Multi-AGV Systems in Autonomous Warehousing Applications », *in: IEEE Transactions on Automation Science and Engineering* 13.4 (Oct. 2016), pp. 1433–1447, ISSN: 1558-3783, DOI: 10.1109/TASE.2016.2603781.
- [DVD20] Matthias De Ryck, Mark Versteyhe, and Frederik Debrouwere, « Automated Guided Vehicle Systems, State-of-the-Art Control Algorithms and Techniques », *in: (2020)*, DOI: 10.1016/j.jmsy.2019.12.002.
- [DVS20] Matthias De Ryck, Mark Versteyhe, and Keivan Shariatmadar, « Resource Management in Decentralized Industrial Automated Guided Vehicle Systems », *in: Journal of Manufacturing Systems* 54 (Jan. 2020), pp. 204–214, DOI: 10.1016/j.jmsy.2019.11.003.

-
- [EBW16] Behzad Esmailian, Sara Behdad, and Ben Wang, « The Evolution and Future of Manufacturing: A Review », *in: Journal of Manufacturing Systems* 39 (Apr. 2016), pp. 79–100, ISSN: 0278-6125, DOI: 10.1016/j.jmsy.2016.03.001.
- [Elm+15] Akram Elmitwally *et al.*, « A Fuzzy-Multiagent Self-Healing Scheme for a Distribution System With Distributed Generations », *in: IEEE Transactions on Power Systems* 30.5 (Sept. 2015), pp. 2612–2622, ISSN: 1558-0679, DOI: 10.1109/TPWRS.2014.2366072.
- [Eró+19] Endre Erős *et al.*, « A ROS2 Based Communication Architecture for Control in Collaborative and Intelligent Automation Systems », *in: Procedia Manufacturing*, 29th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2019), June 24-28, 2019, Limerick, Ireland, Beyond Industry 4.0: Industrial Advances, Engineering Education and Intelligent Manufacturing 38 (Jan. 2019), pp. 349–357, ISSN: 2351-9789, DOI: 10.1016/j.promfg.2020.01.045.
- [ESI17] Salma El-Ansary, Omar Shehata, and Elsayed Imam Morgan, *Airport Management Controller: A Multi-Robot Task-Allocation Approach*, Jan. 2017, p. 30, DOI: 10.1145/3029610.3029623.
- [EU20] EU, *EU Road Safety: Towards "Vision Zero". CINEA Horizon 2020 Publication*, tech. rep., 2020.
- [FAS20] William Ferreira, Fabiano Armellini, and Luis Antonio Santa-Eulalia, « Simulation in Industry 4.0: A State-of-the-Art Review », *in: Computers & Industrial Engineering* (Sept. 2020), p. 106868, DOI: 10.1016/j.cie.2020.106868.
- [FO13] Alain-Jérôme Fougères and Egon Ostrosi, « Fuzzy Agent-Based Approach for Consensual Design Synthesis in Product Configuration », *in: Integrated Computer-Aided Engineering* 20.3 (Jan. 2013), pp. 259–274, ISSN: 1069-2509, DOI: 10.3233/ICA-130434.
- [Fou02] Alain-Jérôme Fougères, « Model of Cognitive Agents to Simulate Complex Information Systems », *in: IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, Oct. 2002, 6 pp. vol.4-, DOI: 10.1109/ICSMC.2002.1173308.
- [Fou13] Alain-Jérôme Fougères, « A Modelling Approach Based on Fuzzy Agents », *in: arXiv:1302.6442 [cs]* (Feb. 2013), arXiv: 1302.6442 [cs].
- [GC11] Serge Guillaume and Brigitte Charnomordic, « Learning Interpretable Fuzzy Inference Systems with FisPro », *in: Information Sciences* 181.20 (Oct. 2011), pp. 4409–4427, ISSN: 00200255, DOI: 10.1016/j.ins.2011.03.025.

-
- [Gei+13] Andreas Geiger *et al.*, « Vision Meets Robotics: The KITTI Dataset », *in: The International Journal of Robotics Research* 32.11 (Sept. 2013), pp. 1231–1237, ISSN: 0278-3649, DOI: 10.1177/0278364913491297.
- [Glo+21] Nicholas Glorio *et al.*, « An Adaptive Approach for the Coordination of Autonomous Vehicles at Intersections », *in: 2021 IEEE 30th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, Oct. 2021, pp. 1–6, DOI: 10.1109/WETICE53228.2021.00012.
- [GM04] Brian Gerkey and Maja Mataric, « A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems », *in: I. J. Robotic Res.* 23 (Sept. 2004), pp. 939–954, DOI: 10.1177/0278364904045564.
- [GT22] Gürkan Gürgöze and İbrahim Türkoğlu, « A Novel, Energy-Efficient Smart Speed Adaptation Based on the Gini Coefficient in Autonomous Mobile Robots », *in: Electronics* 11.19 (Jan. 2022), p. 2982, ISSN: 2079-9292, DOI: 10.3390/electronics11192982.
- [Haf+13] Michael R. Hafner *et al.*, « Cooperative Collision Avoidance at Intersections: Algorithms and Experiments », *in: IEEE Transactions on Intelligent Transportation Systems* 14.3 (Sept. 2013), pp. 1162–1175, ISSN: 1558-0016, DOI: 10.1109/TITS.2013.2252901.
- [Hag04] Hani Hagaras, « Hagaras, H.: A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots. IEEE Transactions on Fuzzy Systems 12(4), 524-539 », *in: Fuzzy Systems, IEEE Transactions on* 12 (Sept. 2004), pp. 524–539, DOI: 10.1109/TFUZZ.2004.832538.
- [Hao+23] Yu Hao *et al.*, « Understanding the Impact of Image Quality and Distance of Objects to Object Detection Performance », *in: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2023, pp. 11436–11442, DOI: 10.1109/IROS55552.2023.10342139.
- [HDA17] Sami Haddadin, Alessandro De Luca, and Alin Albu-Schaffer, « Robot Collisions: A Survey on Detection, Isolation, and Identification », *in: IEEE Transactions on Robotics* 33.6 (Dec. 2017), pp. 1292–1312, ISSN: 1552-3098, 1941-0468, DOI: 10.1109/TR0.2017.2723903.
- [Hen+19] Abdelfetah Hentout *et al.*, « Human–Robot Interaction in Industrial Collaborative Robotics: A Literature Review of the Decade 2008–2017 », *in: Advanced Robotics* 33.15-16 (Aug. 2019), pp. 764–799, ISSN: 0169-1864, DOI: 10.1080/01691864.2019.1636714.

-
- [HIA22] Hasitha Sanjeeva Hewawasam, M. Yousef Ibrahim, and Gayan Kahandawa Apuhamillage, « Past, Present and Future of Path-Planning Algorithms for Mobile Robot Navigation in Dynamic Environments », *in: IEEE Open Journal of the Industrial Electronics Society* 3 (2022), pp. 353–365, ISSN: 2644-1284, DOI: 10.1109/OJIES.2022.3179617.
- [HK13] Ahmed Hussein and Alaa Khamis, *Market-Based Approach to Multi-robot Task Allocation*, Dec. 2013, DOI: 10.1109/ICBR.2013.6729278.
- [HNK12] Tang Hong, Danial Nakhaeinia, and Babak Karasfi, « Application of Fuzzy Logic in Mobile Robot Navigation », *in: Mar. 2012*, ISBN: 978-953-51-0396-7, DOI: 10.5772/36358.
- [HS19] Zool Hilmi Ismail and Nohaidda Sariff, « A Survey and Analysis of Cooperative Multi-Agent Robot Systems: Challenges and Directions », *in: Applications of Mobile Robots*, ed. by Efren Gorrostieta Hurtado, IntechOpen, Mar. 2019, ISBN: 978-1-78985-755-9 978-1-78985-756-6, DOI: 10.5772/intechopen.79337.
- [Hyl11] Paweł Hyla, « Automated Guided Vehicles Navigating Problem In Container Terminal », *in: Logistics and Transport* (Jan. 2011).
- [HZ05] Xiaolin Hu and Bernard P. Zeigler, « A Simulation-Based Virtual Environment to Study Cooperative Robotic Systems », *in: Integrated Computer-Aided Engineering* 12.4 (Jan. 2005), pp. 353–367, ISSN: 1069-2509, DOI: 10.3233/ICA-2005-12404.
- [ICAE23] J. Grosset, A. Ndao, *et al.*, « A Cooperative Approach to Avoiding Obstacles and Collisions between Autonomous Industrial Vehicles in a Simulation Platform », *in: Integrated Computer-Aided Engineering* 30.1 (Jan. 2023), pp. 19–40, ISSN: 1069-2509, DOI: 10.3233/ICA-220694.
- [ICAE24] J. Grosset, A.-J. Fougères, M. Djoko-Kouam, and J.-M. Bonnin, « Multi-Agent Simulation of Autonomous Industrial Vehicle Fleets: Towards Dynamic Task Allocation in V2X Cooperation Mode », *in: Integrated Computer-Aided Engineering* 31.3 (Apr. 2024), pp. 249–266, ISSN: 10692509, 18758835, DOI: 10.3233/ICA-240735.
- [ITS14a] ITS, *Intelligent Transport Systems; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service, ETSI EN 302 637-2 V1.3.2, 2014*. Tech. rep., 2014.

-
- [ITS14b] ITS, *Intelligent Transport Systems; Vehicular Communications; Basic Set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service*, ETSI EN 302 637-3 V1.2.1, tech. rep., 2014.
- [ITS19] ITS, *Intelligent Transport Systems; Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS), Release 2*, ETSI TR 103 562 V2.1.1, tech. rep., 2019.
- [ITS20] ITS, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities Layer Protocols and Communication Requirements for Infrastructure Services*, ETSI TS 103 301 V1.3.1 (2020-02). Tech. rep., 2020.
- [Jia+] Sixian Jia, Yan Xu, *et al.*, « Review of Common Problems and Solutions of Camera Calibration », *in: ()*.
- [Jin+20] Peng Jing *et al.*, « Agent-Based Simulation of Autonomous Vehicles: A Systematic Literature Review », *in: IEEE Access* 8 (2020), pp. 79089–79103, ISSN: 2169-3536, DOI: 10.1109/ACCESS.2020.2990295.
- [JM13] Xiao Jia and Max Meng, *A Survey and Analysis of Task Allocation Algorithms in Multi-Robot Systems*, Dec. 2013, p. 2285, ISBN: 978-1-4799-2744-9, DOI: 10.1109/ROBIO.2013.6739809.
- [Kar+21] Karthik Karur *et al.*, « A Survey of Path Planning Algorithms for Mobile Robots », *in: Vehicles* 3.3 (Sept. 2021), pp. 448–468, ISSN: 2624-8921, DOI: 10.3390/vehicles3030027.
- [Kat+15] Shinpei Kato, Eijiro Takeuchi, *et al.*, « An Open Approach to Autonomous Vehicles », *in: IEEE Micro* 35.6 (Nov. 2015), pp. 60–68, ISSN: 1937-4143, DOI: 10.1109/MM.2015.133.
- [Kat+18] Shinpei Kato, Shota Tokunaga, *et al.*, « Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems », *in: 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, Apr. 2018, pp. 287–296, DOI: 10.1109/ICCPS.2018.00035.
- [Kha+20] Hamid Khayyam *et al.*, *Artificial Intelligence and Internet of Things for Autonomous Vehicles*, 2020, pp. 39–68, ISBN: 978-3-030-18962-4, DOI: 10.1007/978-3-030-18963-1_2.

-
- [KHE15] Alaa Khamis, Ahmed Hussein, and Ahmed Elmogy, « Multi-Robot Task Allocation: A Review of the State-of-the-Art », *in: Studies in Computational Intelligence*, vol. 604, May 2015, pp. 31–51, ISBN: 978-3-319-18299-5, DOI: 10.1007/978-3-319-18299-5_2.
- [KHJ18] Dae Hwan Kim, Nguyen Trong Hai, and Woong Yeol Joe, « A Guide to Selecting Path Planning Algorithm for Automated Guided Vehicle (AGV) », *in: AETA 2017 - Recent Advances in Electrical Engineering and Related Sciences: Theory and Application*, ed. by Vo Hoang Duy *et al.*, vol. 465, Cham: Springer International Publishing, 2018, pp. 587–596, ISBN: 978-3-319-69813-7 978-3-319-69814-4, DOI: 10.1007/978-3-319-69814-4_56.
- [KHW13] Henning Kagermann, Johannes Helbig, and Wolfgang Wahlster, *Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0: Securing the Future of German Manufacturing Industry ; Final Report of the Industrie 4.0 Working Group*, Forschungsunion, 2013.
- [Kop62] Richard E. Kopp, « Pontryagin Maximum Principle », *in: Mathematics in Science and Engineering*, ed. by George Leitmann, vol. 5, Optimization Techniques, Elsevier, Jan. 1962, pp. 255–279, DOI: 10.1016/S0076-5392(08)62095-0.
- [KS19] Qazi Shaheen Kabir and Yoshinori Suzuki, « Comparative Analysis of Different Routing Heuristics for the Battery Management of Automated Guided Vehicles », *in: International Journal of Production Research* 57.2 (Jan. 2019), pp. 624–641, ISSN: 0020-7543, DOI: 10.1080/00207543.2018.1475761.
- [KSD13] G. Ayorkor Korsah, Anthony Stentz, and M. Bernardine Dias, « A Comprehensive Taxonomy for Multi-Robot Task Allocation », *in: The International Journal of Robotics Research* 32.12 (Oct. 2013), pp. 1495–1512, ISSN: 0278-3649, DOI: 10.1177/0278364913496484.
- [Kun+06] Voemir Kunchev *et al.*, *Path Planning and Obstacle Avoidance for Autonomous Mobile Robots: A Review*, Oct. 2006, p. 544, ISBN: 978-3-540-46537-9, DOI: 10.1007/11893004_70.
- [LA23] Anbalagan Loganathan and Nur Syazreen Ahmad, « A Systematic Review on Recent Advances in Autonomous Mobile Robot Navigation », *in: Engineering Science and Technology, an International Journal* 40 (Apr. 2023), p. 101343, ISSN: 2215-0986, DOI: 10.1016/j.jestch.2023.101343.

-
- [Las+14] Heiner Lasi *et al.*, « Industry 4.0 », *in: Business & Information Systems Engineering* 6.4 (Aug. 2014), pp. 239–242, ISSN: 1867-0202, DOI: 10.1007/s12599-014-0334-4.
- [Lau+16] Sven Laux *et al.*, « Demo: OpenC2X — An Open Source Experimental and Prototyping Platform Supporting ETSI ITS-G5 », *in: 2016 IEEE Vehicular Networking Conference (VNC)*, Dec. 2016, pp. 1–2, DOI: 10.1109/VNC.2016.7835955.
- [LBE18] Chaymaa Lamini, Said Benhlima, and Ali Elbekri, « Genetic Algorithm Based Approach for Autonomous Mobile Robot Path Planning », *in: Procedia Computer Science*, PROCEEDINGS OF THE FIRST INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING IN DATA SCIENCES, ICDS2017 127 (Jan. 2018), pp. 180–189, ISSN: 1877-0509, DOI: 10.1016/j.procs.2018.01.113.
- [Lei+15] Tim Leinmüller *et al.*, « Multi-Channel Usage in Day 2 and beyond EU V2X Systems », *in: 2015*.
- [Liu+23] Zhe Liu *et al.*, « Cooperative Platoon Control of Automated Industrial Vehicles: A Synchronization Approach and Real-World Experiments », *in: IEEE/ASME Transactions on Mechatronics* 28.1 (Feb. 2023), pp. 245–256, ISSN: 1941-014X, DOI: 10.1109/TMECH.2022.3193410.
- [LN22] Min-Fan Ricky Lee and Asep Nugroho, « Intelligent Energy Management System for Mobile Robot », *in: Sustainability* 14.16 (Jan. 2022), p. 10056, ISSN: 2071-1050, DOI: 10.3390/su141610056.
- [Lu+20] Qiong Lu *et al.*, « The Impact of Autonomous Vehicles on Urban Traffic Network Capacity: An Experimental Analysis by Microscopic Traffic Simulation », *in: Transportation Letters* 12.8 (Sept. 2020), pp. 540–549, ISSN: 1942-7867, DOI: 10.1080/19427867.2019.1662561.
- [Lu17] Yang Lu, « Industry 4.0: A Survey on Technologies, Applications and Open Research Issues », *in: Journal of Industrial Information Integration* 6 (June 2017), pp. 1–10, ISSN: 2452-414X, DOI: 10.1016/j.jii.2017.04.005.
- [Lya19] Nikita Lyamin, « Performance Evaluation of Safety Critical ITS-G5 V2V Communications for Cooperative Driving Applications », *in: Halmstad University dissertations* 53 (2019).

-
- [Mac16] Charles M. Macal, « Everything You Need to Know about Agent-Based Modelling and Simulation », *in: Journal of Simulation* 10.2 (May 2016), pp. 144–156, ISSN: 1747-7786, DOI: 10.1057/jos.2016.7.
- [Mao+23] Jiageng Mao *et al.*, « 3D Object Detection for Autonomous Driving: A Comprehensive Survey », *in: International Journal of Computer Vision* 131.8 (Aug. 2023), pp. 1909–1963, ISSN: 1573-1405, DOI: 10.1007/s11263-023-01790-1.
- [Mar+19] Helmut Martin *et al.*, « Identification of Performance Limitations of Sensing Technologies for Automated Driving », *in: 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, Nov. 2019, pp. 1–6, DOI: 10.1109/ICCVE45908.2019.8965181.
- [MCZ21] Stefano Mariani, Giacomo Cabri, and Franco Zambonelli, « Coordination of Autonomous Vehicles: Taxonomy and Survey », *in: ACM Computing Surveys* 54.1 (Apr. 2021), pp. 1–33, ISSN: 0360-0300, 1557-7341, DOI: 10.1145/3431231, arXiv: 2001.02443.
- [Mey+18] Andry Meylani *et al.*, *Different Types of Fuzzy Logic in Obstacles Avoidance of Mobile Robot*, IEEE, Piscataway, NJ, USA, Oct. 2018, p. 100, DOI: 10.1109/ICECOS.2018.8605206.
- [MG19] Martijn Mes and Berry Gerrits, « Multi-Agent Systems », *in: Operations, Logistics and Supply Chain Management*, Springer, 2019, pp. 611–636, DOI: 10.1007/978-3-319-92447-2_27.
- [Mİ21] Christopher Medrano-Berumen and Mustafa İlhan Akbaş, « Validation of Decision-Making in Artificial Intelligence-Based Autonomous Vehicles », *in: Journal of Information and Telecommunication* 5.1 (Jan. 2021), pp. 83–103, ISSN: 2475-1839, DOI: 10.1080/24751839.2020.1824154.
- [Mia+21] Changwei Miao *et al.*, « Path Planning Optimization of Indoor Mobile Robot Based on Adaptive Ant Colony Algorithm », *in: Computers & Industrial Engineering* 156 (June 2021), p. 107230, ISSN: 0360-8352, DOI: 10.1016/j.cie.2021.107230.
- [Mis+22] Divyendu Kumar Mishra *et al.*, « Design of Mobile Robot Navigation Controller Using Neuro-Fuzzy Logic System », *in: Computers and Electrical Engineering* 101 (July 2022), p. 108044, ISSN: 0045-7906, DOI: 10.1016/j.compeleceng.2022.108044.

-
- [MK16] Hang Ma and Sven Koenig, « Optimal Target Assignment and Path Finding for Teams of Agents », *in: Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, AAMAS '16, Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, May 2016, pp. 1144–1152, ISBN: 978-1-4503-4239-1.
- [MKA16] Yuya Maruyama, Shinpei Kato, and Takuya Azumi, « Exploring the Performance of ROS2 », *in: Proceedings of the 13th International Conference on Embedded Software*, Pittsburgh Pennsylvania: ACM, Oct. 2016, pp. 1–10, ISBN: 978-1-4503-4485-2, DOI: 10.1145/2968478.2968502.
- [MM30] Mauricio F. Jaramillo Morales and Juan B. Gómez Mendoza, « Mixed Energy Model for a Differential Guide Mobile Robot », *in: 2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*, Miedzyzdroje, Poland, 27/2018-08-30, pp. 114–119, DOI: 10.1109/MMAR.2018.8486117.
- [MN10] Charles M. Macal and Michael North, « Tutorial on Agent-Based Modelling and Simulation », *in: J. Simulation* 4 (Sept. 2010), pp. 151–162, DOI: 10.1057/jos.2010.3.
- [Nas+19] Martzel Nascimento *et al.*, *A Systematic Literature Review about the Impact of Artificial Intelligence on Autonomous Vehicle Safety*, Apr. 2019.
- [NC18] Amir Nasrinahar and Joon Huang Chuah, « Intelligent Motion Planning of a Mobile Robot with Dynamic Obstacle Avoidance », *in: Journal on Vehicle Routing Algorithms* 1 (Jan. 2018), DOI: 10.1007/s41604-018-0007-4.
- [NCP19] Gaurang Naik, Biplav Choudhury, and Jung-Min Park, « IEEE 802.11bd & 5G NR V2X: Evolution of Radio Access Technologies for V2X Communications », *in: IEEE Access* 7 (2019), pp. 70169–70184, ISSN: 2169-3536, DOI: 10.1109/ACCESS.2019.2919489.
- [NY22] Byeongjoon Noh and Hwasoo Yeo, « A Novel Method of Predictive Collision Risk Area Estimation for Proactive Pedestrian Accident Prevention System in Urban Surveillance Infrastructure », *in: Transportation Research Part C: Emerging Technologies* 137 (Apr. 2022), p. 103570, ISSN: 0968-090X, DOI: 10.1016/j.trc.2022.103570.
- [OB20] Ouazna Oukacha and Nicolas Boizot, « Consumption Minimization for an Academic Model of a Vehicle », *in: Optimal Control Applications and Methods* 41.4 (2020), pp. 1288–1320, ISSN: 1099-1514, DOI: 10.1002/oca.2604.
- [Ode+00] James Odell *et al.*, « Extending UML for Agents », *in:* (Sept. 2000).

-
- [OFF12] Egon Ostrosi, Alain-Jérôme Fougères, and Michel Ferney, « Fuzzy Agents for Product Configuration in Collaborative and Distributed Design Process », *in: Applied Soft Computing* 12.8 (Aug. 2012), pp. 2091–2105, ISSN: 1568-4946, DOI: 10.1016/j.asoc.2012.03.005.
- [ÖG31] Tuncer I. Ören and Nasser Ghasem-Aghaee, « Personality Representation Processable in Fuzzy Logic for Human Behavior Simulation », *in: vol. Proceedings of SCSC 2003, Phoenix, AZ, USA, 2003-08-01/2003-07-31*, pp. 3–10.
- [Pan17] Anish Pandey, « Mobile Robot Navigation and Obstacle Avoidance Techniques: A Review », *in: International Robotics & Automation Journal* 2.3 (May 2017), ISSN: 25748092, DOI: 10.15406/iratj.2017.02.00023.
- [Pat+19] Bhumeshwar Patle *et al.*, « A Review: On Path Planning Strategies for Navigation of Mobile Robot », *in: Defence Technology* 15.4 (Aug. 2019), pp. 582–606, ISSN: 2214-9147, DOI: 10.1016/j.dt.2019.04.011.
- [Per+20] Ricardo Silva Peres *et al.*, « Industrial Artificial Intelligence in Industry 4.0 - Systematic Review, Challenges and Outlook », *in: IEEE Access* 8 (2020), pp. 220121–220139, ISSN: 2169-3536, DOI: 10.1109/ACCESS.2020.3042874.
- [PGS19] Jéssica Bruna Perussi, Fernando Gressler, and Robson Seleme, « Supply Chain 4.0: Autonomous Vehicles and Equipment to Meet Demand », *in: 8.4* (2019).
- [PPP09] Saroj Pradhan, Dayal Parhi, and Anup Panda, « Fuzzy Logic Techniques for Navigation of Several Mobile Robots », *in: Applied Soft Computing* 9 (Jan. 2009), pp. 290–304, DOI: 10.1016/j.asoc.2008.04.008.
- [PR17] Ana C. Pereira and Fernando Romero, « A Review of the Meanings and the Implications of the Industry 4.0 Concept », *in: Procedia Manufacturing, Manufacturing Engineering Society International Conference 2017, MESIC 2017, 28-30 June 2017, Vigo (Pontevedra), Spain* 13 (Jan. 2017), pp. 1206–1214, ISSN: 2351-9789, DOI: 10.1016/j.promfg.2017.09.032.
- [Rai+22] Mehak Raibail *et al.*, « Decentralized Multi-Robot Collision Avoidance: A Systematic Review from 2015 to 2021 », *in: Symmetry* 14.3 (Mar. 2022), p. 610, ISSN: 2073-8994, DOI: 10.3390/sym14030610.
- [Ras83] Jens Rasmussen, « Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models », *in: IEEE Transactions on Systems, Man, and Cybernetics* SMC-13.3 (May 1983), pp. 257–266, ISSN: 2168-2909, DOI: 10.1109/TSMC.1983.6313160.

-
- [RAT19] Yara Rizk, Mariette Awad, and Edward W. Tunstel, *Cooperative Heterogeneous Multi-Robot Systems: A Survey*, Jan. 2019.
- [RK22] Ravi Raj and Andrzej Kos, « A Comprehensive Study of Mobile Robot: History, Developments, Applications, and Future Research Perspectives », *in: Applied Sciences* 12.14 (Jan. 2022), p. 6951, ISSN: 2076-3417, DOI: 10.3390/app12146951.
- [RM17] Jackeline Rios-Torres and Andreas A. Malikopoulos, « A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps », *in: IEEE Transactions on Intelligent Transportation Systems* 18.5 (May 2017), pp. 1066–1077, ISSN: 1558-0016, DOI: 10.1109/TITS.2016.2600504.
- [RND10] Stuart J. Russell, Peter Norvig, and Ernest Davis, *Artificial Intelligence: A Modern Approach*, 3rd ed, Prentice Hall Series in Artificial Intelligence, Upper Saddle River: Prentice Hall, 2010, ISBN: 978-0-13-604259-4.
- [Ron+18] Michele Rondinone *et al.*, « V2X Communications for Infrastructure-Assisted Automated Driving », *in: 2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, June 2018, pp. 14–19, DOI: 10.1109/WoWMoM.2018.8449775.
- [Ros09] Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, John Wiley & Sons, Dec. 2009, ISBN: 978-0-470-74851-0.
- [Rüß+15] Michael Rüßmann *et al.*, « Industry 4.0: The Future of Productivity and Growth in Manufacturing », *in: (2015)*.
- [Saf+21] Saeid Safavi *et al.*, « Multi-Sensor Fault Detection, Identification, Isolation and Health Forecasting for Autonomous Vehicles », *in: Sensors* 21.7 (Jan. 2021), p. 2547, ISSN: 1424-8220, DOI: 10.3390/s21072547.
- [Sal+99] Miguel Angel Salichs *et al.*, « Localization System for Mobile Robots in Indoor Environments », *in: Integrated Computer-Aided Engineering* 6.4 (Dec. 1999), pp. 303–318, ISSN: 1069-2509.
- [SBL19] Florian Schiegg, Nadia Brahmi, and Ignacio Llatser, « Analytical Performance Evaluation of the Collective Perception Service in C-V2X Mode 4 Networks », *in: Oct. 2019*, pp. 181–188, DOI: 10.1109/ITSC.2019.8917214.
- [SCO16] SCOOP@f, *Projet SCOOP : Véhicules et Routes Connectés. Messages*, tech. rep., Apr. 2016.

-
- [Seq+19] Luis Sequeira *et al.*, « A Lane Merge Coordination Model for a V2X Scenario », *in: 2019 European Conference on Networks and Communications (EuCNC)*, June 2019, pp. 198–203, DOI: 10.1109/EuCNC.2019.8801987.
- [SGD11] Christoph Sommer, Reinhard German, and Falko Dressler, « Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis », *in: IEEE Transactions on Mobile Computing* 10.1 (Jan. 2011), pp. 3–15, ISSN: 1558-0660, DOI: 10.1109/TMC.2010.133.
- [Shi+18] Aggrey Shitsukane *et al.*, « A Survey on Obstacles Avoidance Mobile Robot in Static Unknown Environment », *in: International Journal of Computer (IJC)* (Mar. 2018).
- [SNS11] Roland Siegwart, Illah Reza Nourbakhsh, and Davide Scaramuzza, *Introduction to Autonomous Mobile Robots, Second Edition*, MIT Press, Feb. 2011, ISBN: 978-0-262-01535-6.
- [Ste+19] Roni Stern *et al.*, *Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks*, June 2019.
- [Šte+21] Alexandr Štefek *et al.*, « Optimization of Fuzzy Logic Controller Used for a Differential Drive Wheeled Mobile Robot », *in: Applied Sciences* 11.13 (Jan. 2021), p. 6023, ISSN: 2076-3417, DOI: 10.3390/app11136023.
- [Tan+18] Zhaoxiang Tan, Shaofeng Lu, *et al.*, « Adaptive Partial Train Speed Trajectory Optimization », *in: Energies* 11.12 (Dec. 2018), p. 3302, ISSN: 1996-1073, DOI: 10.3390/en11123302.
- [TBS18] Naoum Tsolakis, Dimitrios Bechtsis, and Jagjit Singh Srail, « Intelligent Autonomous Vehicles in Digital Supply Chains: From Conceptualisation, to Simulation Modelling, to Real-World Operations », *in: Business Process Management Journal* 25.3 (Jan. 2018), pp. 414–437, ISSN: 1463-7154, DOI: 10.1108/BPMJ-11-2017-0330.
- [TCN21] Andrea Testa, Andrea Camisa, and Giuseppe Notarstefano, « ChoiRbot: A ROS 2 Toolbox for Cooperative Robotics », *in: IEEE Robotics and Automation Letters* 6.2 (Apr. 2021), pp. 2714–2720, ISSN: 2377-3766, 2377-3774, DOI: 10.1109/LRA.2021.3061366, arXiv: 2010.13431.
- [TH19] Joze Tavcar and Imre Horvath, « A Review of the Principles of Designing Smart Cyber-Physical Systems for Run-Time Adaptation: Learned Lessons and Open Issues », *in: IEEE Transactions on Systems, Man, and Cybernetics: Systems*

-
- 49.1 (Jan. 2019), pp. 145–158, ISSN: 2168-2216, 2168-2232, DOI: 10.1109/TSMC.2018.2814539.
- [TM21] Jia Hao Tan and Tariq Masood, « Adoption of Industry 4.0 Technologies in Airports », *in*: (2021).
- [Tsu+19] Manabu Tsukada, Masahiro Kitazawa, *et al.*, « Cooperative Awareness Using Roadside Unit Networks in Mixed Traffic », *in*: *2019 IEEE Vehicular Networking Conference (VNC)*, Dec. 2019, pp. 1–8, DOI: 10.1109/VNC48660.2019.9062773.
- [Tsu+20] Manabu Tsukada, Takaharu Oi, *et al.*, « AutoC2X: Open-source Software to Realize V2X Cooperative Perception among Autonomous Vehicles », *in*: *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, Nov. 2020, pp. 1–6, DOI: 10.1109/VTC2020-Fall149728.2020.9348525.
- [TX21] Hamid Taheri and Zhao Chun Xia, « SLAM; Definition and Evolution », *in*: *Engineering Applications of Artificial Intelligence* 97 (Jan. 2021), p. 104032, ISSN: 0952-1976, DOI: 10.1016/j.engappai.2020.104032.
- [Van+18] Jessica Van Brummelen *et al.*, « Autonomous Vehicle Perception: The Technology of Today and Tomorrow », *in*: *Transportation Research Part C: Emerging Technologies* 89 (Apr. 2018), pp. 384–406, ISSN: 0968-090X, DOI: 10.1016/j.trc.2018.02.012.
- [VLL15] Birgit Vogel-Heuser, Jay Lee, and Paulo Leitão, « Agents Enabling Cyber-Physical Production Systems », *in*: *at - Automatisierungstechnik* 63 (Jan. 2015), DOI: 10.1515/auto-2014-1153.
- [VTC24] Juliette Grosset, Jean-Marie Bonnin, *et al.*, « Generation of V2X Messages from Carla Simulator for Cooperative Perception: Application to Pedestrian Safety. », *in*: *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*, Washington D.C., USA, Oct. 2024.
- [Wan+22] Yuanzhe Wang, Qipeng Liu, *et al.*, « Detection and Isolation of Sensor Attacks for Autonomous Vehicles: Framework, Algorithms, and Validation », *in*: *IEEE Transactions on Intelligent Transportation Systems* 23.7 (July 2022), pp. 8247–8259, ISSN: 1558-0016, DOI: 10.1109/TITS.2021.3077015.
- [Wan+23] Li Wang, Xinyu Zhang, *et al.*, « Multi-Modal 3D Object Detection in Autonomous Driving: A Survey and Taxonomy », *in*: *IEEE Transactions on Intelligent Vehicles* 8.7 (July 2023), pp. 3781–3798, ISSN: 2379-8904, DOI: 10.1109/TIV.2023.3264658.

-
- [Wis22] Yair Wiseman, « Autonomous Vehicles: » *in: Research Anthology on Cross-Disciplinary Designs and Applications of Automation*, ed. by Information Resources Management Association, IGI Global, 2022, pp. 878–889, ISBN: 978-1-66843-694-3 978-1-66843-695-0, DOI: 10.4018/978-1-6684-3694-3.ch043.
- [Woo02] Michael Wooldridge, « An Introduction to MultiAgent Systems », *in: June 2002*, p. 348, ISBN: 978-0-471-49691-5.
- [WT99] David H. Wolpert and Kagan Tumer, *An Introduction to Collective Intelligence*, Aug. 1999, DOI: 10.48550/arXiv.cs/9908014, arXiv: cs/9908014.
- [Xin+21] Haitao Xing *et al.*, « Practical Challenges in CACC Communication: ITS G5, LTE Uu, and LTE Sidelink PC5 », *in: 2021 European Control Conference (ECC)*, June 2021, pp. 1795–1801, DOI: 10.23919/ECC54610.2021.9655171.
- [Xu+21a] Runsheng Xu, Yi Guo, *et al.*, « OpenCDA: An Open Cooperative Driving Automation Framework Integrated with Co-Simulation », *in: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, Sept. 2021, pp. 1155–1162, DOI: 10.1109/ITSC48978.2021.9564825.
- [Xu+21b] Xun Xu, Yuqian Lu, *et al.*, « Industry 4.0 and Industry 5.0—Inception, Conception and Perception », *in: Journal of Manufacturing Systems* 61 (Oct. 2021), pp. 530–535, ISSN: 0278-6125, DOI: 10.1016/j.jmsy.2021.10.006.
- [Xu+23] Runsheng Xu, Hao Xiang, *et al.*, « The OpenCDA Open-Source Ecosystem for Cooperative Driving Automation Research », *in: IEEE Transactions on Intelligent Vehicles* 8.4 (Apr. 2023), pp. 2698–2711, ISSN: 2379-8904, DOI: 10.1109/TIV.2023.3244948.
- [XZ24] Elias Xidias and Paraskevi Zacharia, « Balanced Task Allocation and Motion Planning of a Multi-Robot System under Fuzzy Time Windows », *in: Engineering Computations* 41.5 (Jan. 2024), pp. 1301–1326, ISSN: 0264-4401, DOI: 10.1108/EC-09-2023-0612.
- [Yeo+21] De Jong Yeong *et al.*, « Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review », *in: Sensors* 21.6 (Jan. 2021), p. 2140, ISSN: 1424-8220, DOI: 10.3390/s21062140.
- [Yer+15] Vivek Yerubandi *et al.*, « Navigation System for an Autonomous Robot Using Fuzzy Logic », *in: International Journal of Scientific and Research Publications (IJSRP)* 5 (Feb. 2015), pp. 1–4.

-
- [YJC13] Zhi Yan, Nicolas Jouandeau, and Arab Ali Cherif, « A Survey and Analysis of Multi-Robot Coordination », *in: International Journal of Advanced Robotic Systems* 10.12 (Jan. 2013), p. 399, DOI: 10.5772/57313.
- [YM20] Xiaoyan Yu and Marin Marinov, « A Study on Recent Developments and Issues with Obstacle Detection Systems for Automated Vehicles », *in: Sustainability* 12.8 (Jan. 2020), p. 3281, ISSN: 2071-1050, DOI: 10.3390/su12083281.
- [Yud+19] Hendra Yudha *et al.*, *Performance Comparison of Fuzzy Logic and Neural Network Design for Mobile Robot Navigation*, IEEE, Piscataway, NJ, USA, Oct. 2019, p. 84, DOI: 10.1109/ICECOS47637.2019.8984577.
- [Zha+21] Xiaolong Zhang, Yu Huang, *et al.*, « Optimal Trajectory Planning for Wheeled Mobile Robots under Localization Uncertainty and Energy Efficiency Constraints », *in: Sensors* 21.2 (Jan. 2021), p. 335, ISSN: 1424-8220, DOI: 10.3390/s21020335.
- [Zha+23] Zihan Zhang, Haobing Liu, *et al.*, « Review on the Impacts of Cooperative Automated Driving on Transportation and Environment », *in: Transportation Research Part D: Transport and Environment* 115 (Feb. 2023), p. 103607, ISSN: 1361-9209, DOI: 10.1016/j.trd.2023.103607.
- [Zie+16] Adam Ziebinski *et al.*, « A Survey of ADAS Technologies for the Future Perspective of Sensor Fusion », *in: Computational Collective Intelligence*, ed. by Ngoc Thanh Nguyen *et al.*, Cham: Springer International Publishing, 2016, pp. 135–146, ISBN: 978-3-319-45246-3, DOI: 10.1007/978-3-319-45246-3_13.
- [ZP18] Sorin Eugen Zaharia and Casandra Venera Pietreanu, « Challenges in Airport Digital Transformation », *in: Transportation Research Procedia*, INAIR 2018 Aviation on the Growth Path 35 (Jan. 2018), pp. 90–99, ISSN: 2352-1465, DOI: 10.1016/j.trpro.2018.12.016.
- [ZWC19] Qi-bin Zhang, Peng Wang, and Zong-hai Chen, « An Improved Particle Filter for Mobile Robot Localization Based on Particle Swarm Optimization », *in: Expert Systems with Applications* 135 (Nov. 2019), pp. 181–193, ISSN: 0957-4174, DOI: 10.1016/j.eswa.2019.06.006.

RÉSUMÉ LONG EN FRANÇAIS

Contexte et enjeux

L'avènement de l'industrie 4.0 a introduit de nouveaux défis dans l'optimisation des flux de données, de produits et de matériaux au sein des environnements de production. Les véhicules industriels autonomes (VIAs), y compris les "Automated Guided Vehicles" (AGVs) et autres "Autonomous Mobile Robots" (AMRs), sont apparus comme des solutions prometteuses pour relever ces défis. Cependant, l'introduction et le déploiement de flottes de VIAs dans les entreprises restent problématiques à plusieurs niveaux : l'acceptabilité par les employés, la localisation précise des véhicules, le maintien d'un flux de circulation fluide, la garantie d'une perception fiable dans des environnements dynamiques et en constante évolution, etc. En conséquence, l'autonomie des VIAs est limitée par des trajectoires prédéterminées et des cadres opérationnels rigides.

Cette thèse se concentre sur l'amélioration de **l'autonomie des flottes de VIAs en exploitant des stratégies d'intelligence collective**. En permettant une communication robuste et un échange d'informations de circulation pertinentes entre les VIAs et l'infrastructure, nous visons à améliorer leur adaptabilité, leurs capacités de prise de décision et leur efficacité globale. Les principaux domaines d'intérêt comprennent :

- Adaptation aux contraintes de circulation : améliorer la capacité des VIAs à s'ajuster aux environnements changeants, tels que les zones de stockage dynamiques et les chaînes de production, afin d'assurer un fonctionnement fluide malgré des conditions changeantes. Cette adaptabilité repose sur les avancées des technologies d'intelligence artificielle (IA) et "Internet of Things" (IOT) [Kha+20] pour une meilleure perception de l'environnement.
- Amélioration de la prise de décision : développer des stratégies permettant aux VIAs de prendre des décisions éclairées malgré des informations incomplètes, incertaines ou fragmentées [MI21].
- Communication "Vehicle-To-Everything" (V2X) : faciliter la communication entre les VIAs au sein de la flotte, avec l'infrastructure, et avec les opérateurs humains, améliorant ainsi la coordination et la sécurité.

-
- Efficacité énergétique : mettre en œuvre des stratégies pour réduire la consommation d'énergie des flottes de VIAs, même dans des conditions de circulation et d'exploitation variables [BM16].

Défis techniques

- **Navigation et localisation** : Les VIAs doivent naviguer dans des environnements complexes et dynamiques tout en maintenant une localisation précise. La capacité des VIAs à **percevoir** leur environnement et à éviter les obstacles est fondamentale. En effet, l'**éviterment des obstacles** est une fonction cruciale pour les VIAs, leur permettant de se déplacer en toute sécurité, en particulier dans des environnements complexes et dynamiques avec une circulation dense ou des changements fréquents [RK22]. La fiabilité des capteurs et la robustesse des algorithmes de localisation en temps réel sont essentielles pour une navigation efficace. Le défi consiste à développer des systèmes de perception fiables capables de détecter et de différencier les objets statiques et dynamiques, même dans des conditions défavorables comme une faible luminosité. De plus, les algorithmes de planification de mouvement et d'évitement de collision en temps réel doivent être suffisamment robustes pour gérer les obstacles inattendus sans entraîner de retards opérationnels [HIA22].
- **Perception coopérative** : Inspirée par les avancées dans le secteur automobile, la perception coopérative est essentielle pour améliorer la conscience de la situation parmi les VIAs lors des opérations en flotte. Le défi consiste à développer des protocoles de communication efficaces permettant aux VIAs de partager et d'intégrer des informations sensorielles en temps réel, surtout dans des environnements complexes [Per+20]. Cette coopération est vitale pour assurer une navigation sûre et une exécution efficace des tâches.
- **Allocation des tâches** : L'allocation des tâches est cruciale pour optimiser l'efficacité des flottes de VIAs. Le défi est de concevoir des algorithmes capables d'assigner dynamiquement des tâches en fonction des conditions en temps réel, comme la disponibilité des véhicules, la priorité des tâches et les facteurs environnementaux. S'assurer que ces algorithmes sont évolutifs et adaptables aux exigences opérationnelles variables sans entraîner de temps d'arrêt significatif est une préoccupation majeure [DVD20].
- **Gestion de l'énergie** : La gestion de l'énergie est un élément clé du succès opérationnel des VIAs, influençant directement leur efficacité et leur autonomie. Le

défi est d'optimiser la consommation d'énergie grâce à des algorithmes efficaces et à des technologies de batterie avancées, tout en veillant à ce que les VIAs puissent accomplir leurs tâches sans interruptions fréquentes pour des recharges ou des remplacements de batterie [BM16]. L'équilibre entre l'utilisation d'énergie et les exigences des tâches reste un domaine critique de développement.

Défis opérationnels

- **Intégration avec les systèmes existants** : L'intégration des VIAs dans les systèmes industriels et les flux de travail existants peut être complexe [And+15; BKD18]. Cela nécessite d'harmoniser les opérations des VIAs avec les systèmes hérités, d'ajuster les processus de production et de s'assurer de la compatibilité avec l'infrastructure actuelle. Des stratégies d'intégration efficaces et des conceptions de systèmes adaptables sont nécessaires pour un déploiement réussi [PR17].
- **Communication et coordination** : Les VIAs doivent communiquer et se coordonner efficacement, surtout lorsqu'ils opèrent en flotte. Garantir une communication fiable entre les VIAs et avec les systèmes centraux est un défi, en particulier dans des environnements avec une flotte hétérogène de VIAs. Le développement de protocoles de communication robustes et de stratégies de prise de décision décentralisées est crucial pour une opération sans faille [Per+20].
- **Évolutivité et flexibilité** : À mesure que les besoins industriels évoluent, les systèmes de VIAs doivent être évolutifs et flexibles pour répondre à des exigences opérationnelles variées [DVD20]. Adapter les VIAs à différentes tâches ou lignes de production sans entraîner de temps d'arrêt ou de reconfiguration significatifs représente un défi important. Des conceptions modulaires et adaptables des VIAs peuvent aider à résoudre ce problème.
- **Sécurité et fiabilité** : Assurer la sécurité et la fiabilité des VIAs est essentiel pour prévenir les accidents et maintenir la continuité des opérations. Cela inclut le développement de mécanismes de sécurité, la conformité aux normes de sécurité et la mise en œuvre de systèmes robustes de détection des erreurs et de récupération. La cybersécurité des flottes de VIAs a jusqu'à présent été peu étudiée. Le contexte le plus proche est celui de la cybersécurité automobile. À ce titre, la référence est *ISO/SAE 21434:2021*, qui régit la cybersécurité automobile avec pour objectif de préparer l'arrivée des véhicules connectés et de limiter les risques liés à la cybersécurité des véhicules de transport de passagers. Les "Advanced Driver Assistant

System" (ADAS) tendent à rejoindre la sphère des véhicules autonomes, mais cette norme constitue néanmoins une base pertinente pour établir la cybersécurité de nos flottes de VIAs. De plus, l'utilisation de techniques d'IA pour inférer un niveau de menace cybernétique est une nouvelle approche permettant de couvrir les attaques dites "0-days", c'est-à-dire les attaques qui n'ont jamais été mises en œuvre dans le passé.

Facteurs humains

- **Acceptation et confiance** : L'acceptation et la confiance des opérateurs humains et des parties prenantes envers les VIAs sont cruciales pour une mise en œuvre réussie. La résistance au changement, les inquiétudes concernant la perte d'emploi et le scepticisme quant à la fiabilité des VIAs peuvent freiner leur adoption.
- **Formation et développement des compétences** : Les opérateurs et le personnel de maintenance doivent être correctement formés pour interagir avec les systèmes de VIAs et les gérer. Cela inclut la compréhension de leur fonctionnement, la résolution des problèmes et l'entretien de routine.
- **Coopération avec les robots** : Les cobots, ou robots collaboratifs, sont conçus pour assister les humains dans les tâches, améliorant ainsi la productivité et réduisant les efforts physiques. Cependant, des défis se posent pour garantir une interaction fluide, car cela nécessite des contrôles intuitifs, une communication claire et des protocoles de sécurité solides pour éviter les accidents et les malentendus.

S'attaquer à ces défis nécessite une approche multidisciplinaire, combinant des avancées technologiques avec une prise en compte réfléchie des facteurs opérationnels et humains. En surmontant ces obstacles, le plein potentiel des VIAs peut être réalisé, ce qui conduit à une meilleure efficacité, sécurité et flexibilité dans les environnements industriels.

Questions de recherche

L'intégration des VIAs dans les systèmes industriels modernes présente un éventail de défis, surtout lorsque l'on cherche à améliorer l'autonomie de ces véhicules au sein d'une flotte. Les futurs systèmes robotiques autonomes consisteront probablement en une diversité de robots travaillant ensemble pour accomplir des missions complexes. La

synergie entre ces robots hétérogènes et le potentiel de combiner leurs capacités uniques pourraient apporter des avantages significatifs à une multitude d'applications diverses et étendues [RK22].

Dans ce manuscrit, nous n'aborderons pas les défis humains liés aux VIAs ni les préoccupations concernant la cybersécurité. Nous faisons l'hypothèse forte que les systèmes de communication sont fiables et exempts de menaces cybernétiques. Au lieu de cela, notre attention principale porte sur les défis techniques liés à l'augmentation de l'autonomie des VIAs, notamment par la mise en œuvre de stratégies collectives au sein d'une flotte. L'objectif principal de cette thèse est de proposer et d'évaluer des **stratégies d'intelligence collective** qui peuvent significativement améliorer l'**autonomie et l'efficacité des VIAs** opérant au sein de flottes industrielles. Les questions de recherche suivantes forment la base de cette thèse :

Comment les capacités de navigation et de localisation des VIAs peuvent-elles être améliorées dans des environnements complexes et dynamiques grâce à la perception coopérative et à des stratégies collectives ?

Cette question explore comment permettre aux VIAs de percevoir et de naviguer plus précisément dans leurs environnements. Nous examinerons comment le partage d'informations sensorielles entre les VIAs peut améliorer la conscience de la situation et la précision de la navigation. De plus, nous analyserons comment l'optimisation du traitement local peut conduire à de meilleures performances globales grâce à des stratégies collectives. L'interaction avec l'infrastructure et la perception locale partagée entre robots mobiles au sein d'une flotte sont essentielles pour améliorer l'autonomie globale de la flotte sous des hypothèses de coopération forte.

Quelles stratégies peuvent optimiser l'allocation des tâches au sein d'une flotte de VIAs pour améliorer l'efficacité opérationnelle et réduire les temps d'arrêt ?

Cette question aborde le défi de l'allocation dynamique des tâches. Nous explorerons comment les stratégies collectives peuvent être utilisées pour développer des algorithmes évolutifs qui répartissent efficacement les tâches en fonction des conditions en temps réel et de la disponibilité de la flotte, améliorant ainsi l'efficacité opérationnelle.

Comment la gestion de l'énergie peut-elle être optimisée dans une flotte de VIAs pour prolonger l'autonomie opérationnelle et réduire les interruptions dues à la recharge ou au remplacement des batteries ?

Cette question se concentre sur le défi de la gestion de l'énergie. Nous examinerons des stratégies de gestion collective de l'énergie qui tiennent compte à la fois des niveaux d'énergie individuels des VIAs et des exigences opérationnelles de l'ensemble de la flotte, dans le but de prolonger l'autonomie opérationnelle et de minimiser les perturbations.

Comment gérer plus efficacement l'hétérogénéité des systèmes industriels ?

Cette question cherche à démontrer comment les technologies ITS peuvent être adaptées aux contextes industriels, en permettant un traitement et un contrôle localisés pour améliorer les performances globales.

En effet, une fois les scénarios et les contraintes technologiques de communication identifiés en termes de performances (par exemple, délai, gigue, débit et disponibilité), la sélection de la technologie appropriée dépendra non seulement des caractéristiques techniques mais aussi des développements anticipés et des contraintes industrielles. Le débat et la recherche en cours sur les avantages comparatifs des principales technologies candidates pour les communications V2X dans les environnements routiers [Baz+19] mettent en évidence l'importance d'examiner l'évolution de ces technologies [NCP19] et leur adaptation aux besoins spécifiques des environnements industriels. Cependant, comme mentionné précédemment, nous ne nous concentrerons pas sur la mise en place des technologies de communication.

Quelles méthodologies peuvent être utilisées pour évaluer et affiner rigoureusement les stratégies collectives visant à accroître l'autonomie des flottes de VIAs ?

Cette question porte sur la nécessité d'outils et de méthodologies robustes pour évaluer et optimiser les stratégies collectives dans les flottes de VIAs. Développer l'autonomie des VIAs nécessite une méthode de travail pertinente. Identifier des algorithmes réutilisables ou adaptables pour relever les différents défis de l'augmentation de l'autonomie n'est qu'une partie de la solution. Il est également essentiel de modéliser, simuler, tester et expérimenter les solutions proposées pour en assurer l'efficacité et la viabilité pratique. Nous explorerons comment les environnements de simulation et les techniques de modélisation, telles que les systèmes multi-agents, peuvent être utilisés pour tester et affiner ces stratégies dans diverses conditions. De plus, nous examinerons l'intégration des approches

d'ingénierie logicielle centralisées et décentralisées pour soutenir le développement et la validation de ces stratégies collectives. Cette enquête est essentielle pour décomposer systématiquement les défis complexes d'autonomie auxquels les VIAs sont confrontés et s'assurer que les solutions proposées sont efficaces dans des environnements réels et dynamiques.

De plus, notre méthodologie implique de décomposer le problème de l'autonomie en tâches techniques gérables, en tenant compte de la complexité inhérente des VIAs, qui naviguent, perçoivent, planifient les différentes tâches et gèrent l'énergie dans des environnements dynamiques. Chacun de ces défis techniques sera abordé individuellement dans les différents chapitres qui constituent les contributions de cette thèse. En traitant systématiquement ces tâches et en appliquant des stratégies collectives, nous visons à améliorer la performance et l'autonomie globales des flottes de VIAs.

Organisation de la thèse

Les recherches présentées dans ce manuscrit s'articulent autour de quatre fonctions techniques clés qui améliorent l'autonomie d'un VIA :

- Évitement d'obstacles
- Allocation des tâches
- Perception coopérative
- Gestion de l'énergie

Chapitre 1 : Ce chapitre fournit un aperçu complet de l'état de l'art des AMRs dans le contexte de l'industrie 4.0. Il commence par introduire l'intégration des AMRs dans les environnements industriels et décrit les fonctions principales qu'ils réalisent, telles que l'évitement d'obstacles, l'allocation des tâches, la perception coopérative et la gestion de l'énergie. Le chapitre explore ensuite diverses méthodologies et outils utilisés pour simuler et améliorer ces fonctions, en soulignant l'importance d'améliorer à la fois l'autonomie individuelle et collective. Il passe également en revue les émulateurs et simulateurs existants, en mettant en lumière leur rôle pour combler le fossé entre les simulations et les applications réelles.

Chapitre 2 : Ce chapitre traite du développement de **stratégies collectives d'évite-ment d'obstacles** pour les VIAs dans des environnements industriels complexes. Il commence par introduire une stratégie collective pour la détection des collisions

[ARCI22; ICAE23], suivie d’une solution proposée pour l’évitement global des obstacles en utilisant des approches collaboratives entre les VIAs [ASPAI22]. Le chapitre est également lié à une méthode d’estimation des positions des VIAs dans un cadre industriel fermé, détaillée dans l’annexe C, et se termine par l’émulation d’un scénario d’intersection pour tester les stratégies proposées, ouvrant ainsi la voie à des validations dans le monde réel [ARAC24].

Chapitre 3 : Ce chapitre se concentre sur le développement de **stratégies collectives d’allocation des tâches** pour les VIAs dans des environnements industriels complexes. Il commence par introduire une stratégie d’allocation dynamique des tâches qui exploite la coopération V2X, soulignant la nécessité d’une communication et d’une coordination efficaces pour faciliter les processus collectifs d’allocation des tâches. Le chapitre applique ensuite ces stratégies à un scénario pratique d’entrepôt, simulant la performance des flottes de VIAs dans des conditions dynamiques pour vérifier l’efficacité des approches proposées. Enfin, il aborde les défis posés par des agents défaillants dans l’infrastructure, proposant un début de solution pour détecter et atténuer leur impact, améliorant ainsi la robustesse des stratégies collectives d’allocation des tâches. Ce chapitre est lié à une contribution publiée dans le journal ICAE [ICAE24].

Chapitre 4 : Ce chapitre explore les **stratégies de perception coopérative** pour les VIAs, en soulignant l’importance d’une perception environnementale précise grâce aux systèmes coopératifs. Il commence par discuter de la manière dont l’intégration de données provenant de plusieurs capteurs et des communications V2X peut améliorer la conscience de la situation et la prise de décision au sein des flottes de VIAs. Le chapitre se concentre ensuite sur l’acquisition et l’utilisation de données routières provenant du C-ITS pour améliorer la compréhension de l’environnement dans des environnements industriels dynamiques. Enfin, il présente une architecture pour générer des messages V2X dans le simulateur Carla, visant à améliorer la sécurité des piétons et la perception coopérative, en particulier aux intersections [VTC24].

Chapitre 5 : Ce chapitre traite du développement et de la mise en œuvre de **stratégies collectives de gestion de l’énergie** pour les VIAs, en se concentrant sur les processus de recharge pour améliorer la performance globale de la flotte. Le chapitre commence par introduire une simulation basée sur des agents flous, puis présente un modèle de décision flou spécifiquement conçu pour la recharge des batteries [AISYS24; ALGO24; ASPAI24]. En appliquant la logique floue, nous visons à gérer les incertitudes associées

au processus de recharge. Le modèle de décision flou est appliqué dans un scénario réel, simulant la gestion autonome de la recharge des batteries dans un environnement aéroportuaire, où les scénarios de simulation se basent sur un modèle de charge/décharge proposé pour une batterie d'un VIA. Dans un premier temps, nous comparons les modèles de logique floue avec des approches basées sur des seuils [ASPAI24]. Ensuite, pour créer un cadre de simulation plus réaliste, le modèle énergétique des VIAs est affiné et un modèle de contrôle optimal est établi pour améliorer la consommation d'énergie de chaque VIA au cours de sa mission, tel que détaillé dans l'**annexe D**. Ce travail est actuellement en cours d'examen [ALGO24]. Le chapitre se conclut par l'intégration de la gestion de l'énergie et de l'allocation des tâches, illustrant l'interdépendance de ces deux aspects pour assurer des opérations continues et minimiser les temps d'arrêt [AISYS24].

Conclusion : Cette thèse est le journal de bord d'un voyage à travers le domaine complexe des stratégies d'intelligence collective pour les VIAs, avec un accent sur l'amélioration des fonctionnalités clés pour renforcer l'autonomie des VIAs individuels et la performance globale des flottes de VIAs. Tout au long de ce travail, nous avons exploré de nombreux défis et opportunités inexploitées, dans le but de rendre les VIAs plus capables d'opérer de manière autonome tout en garantissant que les flottes puissent travailler ensemble plus efficacement et en toute sécurité. Il est devenu clair que les méthodes actuelles ont des limites, en particulier pour exploiter pleinement l'intelligence collective afin d'optimiser ces fonctions critiques. Ce dernier chapitre résume les étapes clés de ce manuscrit, en soulignant comment elles ouvrent de nouvelles perspectives pour des explorations futures visant l'objectif que nous avons poursuivi au cours des trois dernières années : développer des stratégies qui non seulement augmentent l'autonomie des VIAs individuels, mais leur permettent également de fonctionner comme une flotte cohérente et intelligente dans des environnements industriels.

Ce dernier chapitre résume les étapes clés de ce manuscrit, en soulignant comment elles ouvrent de nouvelles perspectives pour des explorations futures visant l'objectif que nous avons poursuivi au cours des trois dernières années : développer des stratégies qui non seulement augmentent l'autonomie des AIVs individuels, mais leur permettent également de fonctionner comme une flotte cohérente et intelligente dans des environnements industriels.

Titre : Stratégies d'Intelligence Collectives pour des Véhicules Industriels Autonomes Efficaces

Mot clés : Industrie 4.0, Contexte ITS, Autonomie Coopérative, Systèmes Distribués

Résumé : Les défis de l'industrie 4.0 comprennent l'optimisation des flux de données et la décentralisation de la prise de décision, là où les systèmes centralisés deviennent souvent inadéquats. Les véhicules industriels autonomes (VIAs) doivent devenir plus intelligents et coopératifs, en échangeant des données pertinentes sur le trafic entre eux et avec l'infrastructure. Cette thèse vise à renforcer l'autonomie des VIAs à travers des stratégies d'intelligence collective, améliorant leur adaptabilité, prise de décision et efficacité grâce à une communication et un partage d'informations facilités. Notre méthodologie, basée sur la modélisation, la simulation et des tests scénarisés, vise à proposer des stratégies collectives spécifiques pour renforcer les différentes fonctionnalités clés des VIAs. Tout d'abord,

nous avons amélioré un algorithme d'évitement d'obstacles et développé une stratégie globale basée sur la perception partagée. Nous avons adapté et proposé des messages standardisés au contexte de l'industrie 4.0 et introduit un système de (ré)allocation dynamique des tâches en environnements décentralisés. Ces contributions se basant sur l'hypothèse forte de la perception coopérative, nous avons ensuite proposé une architecture pour générer des données V2X. Enfin, nous avons développé des stratégies collectives de gestion de l'énergie pour chaque VIA basé sur un modèle de décision flou, afin qu'ils puissent déterminer de manière autonome les moments optimaux de recharge et ainsi réduire leurs temps d'inactivité au sein de la flotte.

Title: Collective Intelligence Strategies for Efficient Autonomous Industrial Vehicles

Keywords: Industry 4.0, ITS Context, Cooperative Autonomy, Distributed Systems

Abstract: The challenges of Industry 4.0 include optimizing data flows and decentralizing decision-making, where centralised systems often become inadequate. Autonomous Industrial Vehicles (AIVs) need to become smarter and more cooperative by exchanging relevant traffic data among themselves and with the infrastructure. This thesis aims to enhance the autonomy of AIVs through collective intelligence strategies, improving their adaptability, decision-making, and efficiency by facilitating communication and information sharing. Our methodology, based on modeling, simulation, and scenario testing, seeks to propose specific collective strategies to

strengthen various key functionalities of AIVs. First, we improved an obstacle avoidance algorithm and developed a global strategy based on shared perception. We adapted and proposed standardized messages for the Industry 4.0 context and introduced a dynamic task (re)allocation system in decentralized environments. Based on the strong assumption of cooperative perception, we then proposed an architecture for generating V2X data. Finally, we developed collective energy management strategies for each AIV using a fuzzy decision model, allowing them to autonomously determine the optimal recharge times and thus reduce their downtime within the fleet.